

IN THE UNITED STATES DISTRICT COURT  
FOR THE DISTRICT OF DELAWARE

EBUDDY TECHNOLOGIES B.V.,

§

*Plaintiff,*

§

V.

§

LINKEDIN CORPORATION,

§

*Defendant.*

§

C.A. No. 20-cv-1501-MN

JURY TRIAL DEMANDED

**FIRST AMENDED COMPLAINT FOR PATENT INFRINGEMENT**

Plaintiff, eBuddy Technologies B.V. (hereinafter, “EBT” or “Plaintiff”), by and through its undersigned counsel, hereby respectfully files this First Amended Complaint against Defendant, LinkedIn Corporation (hereinafter, “LinkedIn” or “Defendant”), as follows:

**PARTIES**

1. Plaintiff eBuddy Technologies B.V. is a private limited liability company incorporated under the laws of the Netherlands.
2. Upon information and belief, Defendant LinkedIn is a corporation organized and existing under the laws of the State of Delaware, with a place of business at 2029 Stierlin Court, Mountain View, California 94043, and can be served through its registered agent, Corporation Service Company, 251 Little Falls Drive, Wilmington, Delaware 19808.

**NATURE OF THE ACTION**

3. This is a civil action for patent infringement to stop Defendant’s infringement of United States Patent Nos. 8,510,395 (the ““395 Patent”; attached hereto as Exhibit 1), 9,584,453 (the ““453 Patent”; attached hereto as Exhibit 2), 8,230,135 (the ““135 Patent”; attached hereto as Exhibit 3), and 8,402,179 (the ““179 Patent”; attached hereto as Exhibit 4) (collectively, the “Patents-in-Suit”).

4. EBT alleges that LinkedIn directly and indirectly has infringed and/or continues to infringe the Patents-in-Suit by, *inter alia*, making, using, offering for sale, selling, importing, using (including in connection with internal uses and/or demonstrations), and/or inducing such actions, including in connection with providing the infringing products and instructions/specifications for their use, including as detailed herein.

5. LinkedIn has had actual and/or constructive notice of the infringements alleged herein, including as detailed herein.

6. EBT seeks damages and other relief for LinkedIn's infringement of the Patents-in-Suit, including as detailed herein.

#### **JURISDICTION AND VENUE**

7. This action arises under the Patent Laws of the United States, 35 U.S.C. § 1 *et seq.*, including 35 U.S.C. §§ 271, 281, 283, 284, and 285. This Court has subject matter jurisdiction over this case for patent infringement under 28 U.S.C. §§ 1331 and 1338(a).

8. This Court has personal jurisdiction over LinkedIn, including because LinkedIn is a Delaware corporation; LinkedIn has minimum contacts within the State of Delaware; LinkedIn has purposefully availed itself of the privileges of conducting business in the State of Delaware; LinkedIn regularly conducts business within the State of Delaware; and Plaintiff's cause of action arises directly from LinkedIn's business contacts and other activities in the State of Delaware, including at least by virtue of LinkedIn's infringing methods, systems, computer-readable media, and products, which have been, and are currently, at least practiced, made, and/or used in the State of Delaware. Defendant is subject to this Court's specific and general personal jurisdiction, pursuant to Constitutional Due Process and the Delaware Long Arm Statute. Defendant is subject to this Court's general personal jurisdiction due at least to its continuous and systematic business contacts in Delaware, including related to operations

conducted in Delaware and the infringements alleged herein. Further, on information and belief, LinkedIn is subject to this Court's specific jurisdiction, including because LinkedIn has committed patent infringement in the State of Delaware, including as detailed herein. In addition, LinkedIn induces infringement of the Patents-in-Suit by customers and/or infringing users located in Delaware. Further, on information and belief, LinkedIn regularly conducts and/or solicits business, engages in other persistent courses of conduct, and/or derives substantial revenue from goods and services provided to persons and/or entities in Delaware.

9. Venue is proper in this District pursuant to 28 U.S.C. §§ 1391 and 1400(b), including because LinkedIn resides in the State of Delaware at least by virtue of the fact that it is incorporated in this state and at least some of the direct and/or indirect infringement of the Patents-in-Suit occurs in this District.

#### **THE PATENTS-IN-SUIT**

10. EBT is the owner of all right, title, and interest in the Patents-in-Suit, including the right to sue for past, present, and future infringement thereof and to collect damages for any such past, present, or future infringement. On information and belief and to the extent necessary and at all relevant times, LinkedIn has had at least constructive notice of the Patents-in-Suit. See 35 U.S.C. § 287. The inventions disclosed and claimed in the '395 and '453 Patents comprising, *inter alia*, contact aggregation between different messaging services, provide numerous benefits over any prior systems or methods. The inventions disclosed and claimed in the '135 and '179 Patents comprising, *inter alia*, event notification, provide numerous benefits over any prior systems, methods, or non-transitory computer-readable media.

#### **A. The Claims Of The '395, '453, '135, And '179 Patents Are Directed To Patentable Subject Matter**

11. Including as set forth in the DECLARATION OF DR. RAJEEV SURATI (attached hereto as Exhibit 5), which is incorporated herein by reference as if fully set forth herein, the

claims of the ‘395, ‘453, ‘135, and ‘179 patents are directed to patentable subject matter. Further, for ease of reference and clarity, the matters asserted in said Declaration are likewise asserted herein.

**B. The ‘395 And ‘453 Patents**

**1. *Overview of the ‘395 and ‘453 Patents***

12. The ‘395 Patent is entitled “Contact List Display System And Method.” Application No. 12/774,700, filed on May 5, 2010 and which issued as U.S. Patent No. 8,510,395 is a continuation of Application No. 11/637,316 (filed on December 11, 2006) and claims priority to Provisional Application No. 60/748,988 filed on December 9, 2005. The ‘395 Patent issued on August 13, 2013. The earliest non-provisional application in its priority chain is Application No. 11/637,316, filed on December 11, 2006.

13. The ‘453 Patent is entitled “Contact List Display System And Method.” Application No. 13/941,354, filed on July 12, 2013 and which issued as U.S. Patent No. 9,854,453, is a continuation of Application No. 12/774,700 (which issued as the ‘395 Patent) which was filed May 5, 2010 and which claims priority to provisional and non-provisional filings dated as far back as December 9, 2005. The ‘453 Patent issued on August 13, 2013. The earliest non-provisional application in its priority chain is Application No. 11/637,316, filed on December 11, 2006.

14. As the ‘453 Patent is a continuation of the application for the ‘395 Patent, the specification of the ‘453 Patent is essentially identical to the shared specification of the ‘395 Patent. Thus, for ease of reference, most specification references herein to either the ‘395 or ‘453 Patent will apply equally to the other.

**a. *Overview of Prosecution of the ‘395 and ‘453 Patents***

15. During prosecution of the ‘395 Patent, and as part of Appl. No. 11/637,316, on May

18, 2009, the patent examiner rejected then pending claims 1-20 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 7,496,379 (“Kaplan”).

16. On August 8, 2009, the applicants amended the claims and argued that the cited Kaplan reference did not render the claims, as amended, unpatentable.

17. On November 5, 2009, the examiner issued a final rejection maintaining his rejection of then pending claims 1-20 in view of Kaplan.

18. The applicants filed continuation application, Appl. No. 12/774,700 on May 5, 2010.

19. On April 20, 2011, the examiner issued a final rejection rejecting then pending claims 1-20 under 35 U.S.C. 102(e) as being anticipated by Kaplan.

20. On July 19, 2011, the applicants again amended the claims and argued that the cited Kaplan reference did not render the claims, as amended, unpatentable. However, in an advisory action dated July 26, 2011, the examiner did not enter the proposed amendments, claiming that they raised new matter.

21. On September 16, 2011, applicants filed a request for continued examination (“RCE”). As part of the RCE filing, the previously unentered amendments were entered.

22. On October 16, 2012, the examiner rejected then pending claims 1-8, 10-16 and 18-20 under 35 U.S.C. 103(a) as being unpatentable over the cited Kaplan reference.

23. On January 15, 2013, applicants amended the claims and traversed the examiner’s rejections. On February 7, 2013, the examiner issued a notice of allowance.

24. On February 21, 2013, the applicants filed an RCE and submitted therewith an information disclosure statement. On March 3, 2013, the examiner issued a second notice of allowance.

25. During prosecution of the ‘453 Patent, on July 17, 2015, the patent examiner rejected then pending claims 1-20 under 35 U.S.C. § 101 as being directed to non-statutory subject

matter. The examiner also rejected then pending claims 1-20 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 7,496,379 (“Kaplan”).

26. On November 17, 2015, the applicants amended the claims and argued that they were not unpatentable in view of 35 U.S.C. § 101 and that the cited prior art did not anticipate the amended claims.

27. On February 25, 2016, the patent examiner issued a final office action again rejecting the then pending claims under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The examiner also rejected then pending claims 1-20 under 35 U.S.C. § 102(e) as being anticipated by Kaplan, but on new grounds. The patent examiner also rejected the claims for non-statutory double patenting and obviousness-type double patenting over claims 1-11 of the ‘395 Patent.

28. On April 25, 2016, the applicants filed an response after final amending the claims and arguing were not unpatentable in view of 35 U.S.C. § 101 and that the cited prior art did not anticipate the amended claims.

29. On May 10, 2016, the patent examiner issued an advisory action maintaining his prior rejections under 35 U.S.C. § 101 and 35 U.S.C. § 102, as well as his rejections for double patenting.

30. On August 24, 2016, applicants and the patent examiner had an interview “during which the Examiner and Applicants’ representative discussed the rejections of claims 1-12. More specifically, possible amendments to claim 7 were discussed to overcome the rejections of claims 7-12 under 35 U.S.C. § 101. It was agreed that the above amendments would overcome the rejections of claims 7-12 under 35 U.S.C. § 101. Additionally, it was agreed that a terminal disclaimer would overcome the double patenting rejections with respect to claims 1-12.”

31. On August 25, 2016, applicants filed a terminal disclaimer and also amended the

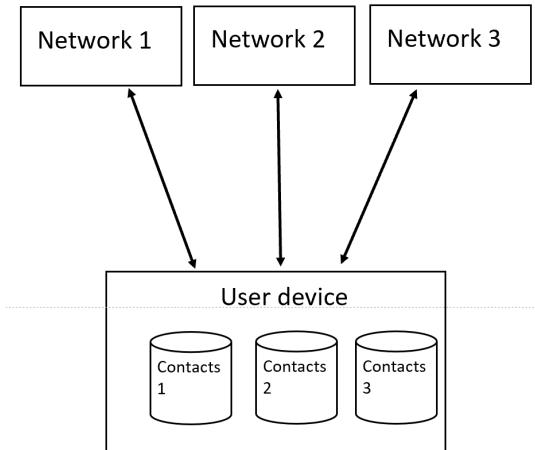
claims and argued that the then pending claims were not unpatentable in view of 35 U.S.C. § 101 and that the cited prior art did not anticipate the amended claims. Specifically, the applicants “amended claims 1, 2, 4, and 7 as proposed by the Examiner, and cancelled claims 9 and 13-20. New claims 21-25 provide non-transitory computer readable medium claims consistent with the allowable method claims 7, 8, and 10-12.”

32. On September 16, 2016, the patent examiner issued a notice of allowance.

b. *Overview of the Unconventional ‘395 and ‘453 Patented Inventions and the Conventional Technology at the Time*

33. At the time of the ‘395 and ‘453 patented inventions, two main forms of direct electronic communication were electronic mail and instant messaging. Ex. 5 at ¶ 42. Generally, instant messaging differed from electronic mail in that conversations through instant messaging could happen in real time and often included presence information, including indications regarding whether a specific user was online and available to chat and real-time updates as to whether a user was typing and what the user typed. ‘395/1:25-33; Ex. 5 at ¶ 42. A user’s corresponding list of other users comprises that user’s contact list. ‘395/1:30; Ex. 5 at ¶ 42. Instant messaging systems further included the ability for users to set status messages, similar to a telephone answering machine. ‘395/1:36-39; Ex. 5 at ¶ 42.

34. Instant messaging systems at the time of the ‘395 and ‘453 Patents comprised a contact list. ‘395/1:28-30; Ex. 5 at ¶ 43. An example of conventional architecture, if a user actually used multiple messaging services, is as follows:



35. Maintenance of a conventional user's contact list(s), including maintenance of an aggregated contact list across multiple messaging services, if done at all, was typically a manual process which could be cumbersome and time consuming, including because each instant messaging service used its own proprietary protocols and formats for maintaining contact lists. Ex. 5 at ¶ 44. A user could add a contact to their contact list in a particular messaging system by, for example, locating the other contact in the messaging system and specifically requesting that user's contact information. Ex. 5 at ¶ 44. Including because each different messaging system has its own contact list, it was particularly cumbersome to add contacts from one messaging service to another where a contact was not an existing member of a particular messaging service. Ex. 5 at ¶ 44. In that case, a user may not be able to add the contact at all or may have to take additional steps to request that the contact create an account or otherwise register with the particular messaging service before being added to the contact list. Ex. 5 at ¶ 44. These proprietary protocols and formats were typically not compatible or transferrable from one messaging system to another. Ex. 5 at ¶ 44. Thus, a user may be required to manually add contacts from one messaging service to another messaging service in order to maintain contacts, including between different messaging systems. Ex. 5 at ¶ 44.

36. At the time of the '395 and '453 inventions reflected in the issued claims (the

“patented inventions”), the most common, conventional and practical way to aggregate contacts was by manually typing each name and/or e-mail address (and/or other contact information) into the user’s contact list, or search for and then pick the contact, who would then later be notified of the request and then could choose to opt in. Ex. 5 at ¶ 45. This often required users to install and maintain multiple, separate local – and often closed and proprietary – applications which required cumbersome, manual processes for maintaining consistent aggregate contact lists across devices. Ex. 5 at ¶ 45. Further, the closed, proprietary local applications often did not even permit the user to access any file containing a list of contacts or otherwise manually transfer contact lists between devices. Ex. 5 at ¶ 45. Nor could the user be guaranteed that an aggregate contact list file from one device would be compatible on a second device. Ex. 5 at ¶ 45. Thus, this often resulted in the user having to manually enter the same contacts on multiple devices for the various different messaging systems the user used. Ex. 5 at ¶ 45.

37. The “SUMMARY” section of the ‘395 and ‘453 Patents states, in part, as follows:

A technique for contact list aggregation across a plurality of different networks involves logging into low level networks through a high level network. A system constructed according to the technique may include a network interface coupled to the different low level networks. The system may further include a contact aggregation engine coupled to the network interface and a network contacts database. In operation the system logs into one or more of the low level networks (or facilitates login for a user). The network contacts database may include some information about contacts associated with the networks from, by way of example but not limitation, previous logins or data explicitly entered by a user. To the extent that the data in the network contacts database is not current, the contact aggregation engine updates the networks contacts database contact information, then provides an aggregated contact list including the contact information to a display device.

A method according to the technique may include logging into a high level network and displaying contacts from the one or more low level networks in an aggregated contact list. The method may further include logging into the one or more low level networks.

‘395/1:61-2:16; Ex. 5 at ¶ 46.

38. The ‘395 and ‘453 claimed inventions have advantages over conventional systems

and methods, including that they allow a user's contact list from multiple instant messaging services to be aggregated into a single contact list and allows the aggregated contact list to be consistently updated in real time across the user's devices, including from a network contacts database. '395/8:59-67; Ex. 5 at ¶ 47. In some embodiments, this includes aggregation of contacts from a high-level messaging service and one or more low level messaging services. '395/8:59-67; Ex. 5 at ¶ 47.

39. At the time of the '395 and '453 patented inventions, there were no iPhones (which first appeared in 2007) or Android devices (which first appeared in 2008), and one of the most commonly used mobile devices was BlackBerry phones. Ex. 5 at ¶ 48. Thus, it was uncommon for most mobile devices used for messaging to have screen sizes of more than about two inches, like those of BlackBerry and Nokia devices. Ex. 5 at ¶ 48. In fact, the shared specification specifically notes that “[t]he information available on a mobile phone display is less than that of a full screen display, such as is available on, for example, a laptop display.” '395/7:44-46; Ex. 5 at ¶ 48. Additionally, mobile devices saw limited, fragmented usage, with most people having only a (non-mobile) desktop computer at home or the office. Ex. 5 at ¶ 48. Furthermore, at that time few, if any, and certainly not widely used, mobile phones allowed multiple applications such as messaging applications to be open or allowed switching between such applications. Ex. 5 at ¶ 48.

40. Moreover, at that time, in order to be able to install applications on a device, especially a work device, the user would need administrative rights on that device. Ex. 5 at ¶ 49. For most users using a work device or someone else's device, for example, one at a school or library, that would not be the case, including because the owner of the device could both prohibit the installation of applications and prohibit certain types of network communications. Ex. 5 at ¶ 49. A server-based application, including the inventions described in the '395 and '453 Patents,

avoids both of these limitations by operating using a network connection, for example, a web connection, and, for example, a browser (which the user's employer, library or school had to allow) not requiring installation of a local, standalone messaging application. Ex. 5 at ¶ 49.

41. Further limitations on network and Internet availability existed at the time, where broadband was growing, but not ubiquitous, and thus the expectation one's network services available to a computer at the time were often insufficient to reliably support continuous connectivity and high bandwidth network activity. Ex. 5 at ¶ 50. Similarly, for mobile devices, mobile data and internet connectivity was not ubiquitous as it is today. Ex. 5 at ¶ 50. At that time, a user would be motivated to minimize mobile data usage because of high cost and limited bandwidth. Ex. 5 at ¶ 50. Nor would a messaging service provider typically have considered having a user use their mobile device or mobile data for web-based applications. Ex. 5 at ¶ 50. As a result, it was conventional to store, example, contact lists, including aggregated contact lists, locally rather than at a network server or web server, or across multiple, different systems. Ex. 5 at ¶ 50.

42. As a result, as noted above, for those conventional messaging applications which provided only a singular service, these limitations were less problematic due to the singular service connections. Ex. 5 at ¶ 51. However, due to these technological limitations, the convention for multi-service local applications was to locally store contact information and serve only as a hand-off between the user's device and each separate messaging service. Ex. 5 at ¶ 51. The patented inventions are unconventional in this regard because they went against the conventional method of locally storing the information for each of the user's messaging services and corresponding contact lists, and instead stored them on the server for retrieval by the user, regardless of where or how the user accessed the web application. Ex. 5 at ¶ 51.

43. There were also no network independent app stores which provided ease of

distribution, so instant messaging services had to make separate applications coded for each type of device they wanted to support and they often had to certify them per mobile network provider. Ex. 5 at ¶ 52. Thus, the conventional way of providing mobile versions of desktop applications would have been to simply make a version of the application for each type of mobile device and get them onto each wireless providers' network. Ex. 5 at ¶ 52. This was incredibly difficult because this was a highly fragmented market, and the limited availability of inexpensive mobile data – meant that when any messaging service updated its protocols, it would require the user take the cumbersome steps of loading or installing the newer version for the user to be able to continue use of the messaging service. Ex. 5 at ¶ 52.

44. At that time, people often transitioned or moved between their computers (for example, between a work or office computer and a home computer), and browsers providing a rich standardized user interface such as AJAX were still relatively new, meaning there was little, if any, standardization across browsers. Ex. 5 at ¶ 53.

45. Consequently, the convention at the time was to create and code your own application, rather than relying on any web-based application especially if the underlying application required server push technology. Ex. 5 at ¶ 54. However, even this was unconventional at the time, including due to the limited resources available on mobile devices. Ex. 5 at ¶ 54. As the convention was separate, proprietary applications, this meant that a user was required to install a different application for each messaging system they used. Ex. 5 at ¶ 54. The ‘395 and ‘453 patented inventions took an unconventional approach to this by providing a single “one-stop-shop” for all of the user’s messaging systems accessible through a single, non-installed web application. Ex. 5 at ¶ 54.

46. Network-, server- and/or web-based applications that worked with multiple messaging services, rather than locally installed client applications for messaging services, were

unconventional at this time, including in view of the conventional technology available described herein, and further including because of the noted limited resources of conventional technology and focus of conventional messaging service providers. Ex. 5 at ¶ 55. Through use of the claimed inventions of the ‘395 and ‘453 Patents, devices with a browser, regardless of which browser, could be used to access your favorite messaging networks, as well as ensure that all of the user’s contacts for each of these networks were available from any device. Ex. 5 at ¶ 55.

47. Local clients or applications required the user to periodically update the client or application. Ex. 5 at ¶ 56. By using the network-based methods and systems described in the ‘395 and ‘453 Patents, any updates could be done at the network or server and the user was ensured they were always accessing the latest version and protocols via the web application. Ex. 5 at ¶ 56. This is due to the server-based operation of the ‘395 and ‘453 patented inventions. ‘395/Figs. 1 & 2; 2:49-64; 3:10-51; 5:1-36; Ex. 5 at ¶ 56. System 100 provides the web application for the claimed multi-service messaging network. ‘395/2:49-50; Ex. 5 at ¶ 56.

48. In one embodiment, the ‘395 and ‘453 patent inventions comprise server 104 in communication with IM server 106 for each messaging service and client 114-N for each of the users of the multi-service system. ‘395/2:50-64; Ex. 5 at ¶ 57. Server 104 may comprise a single server or a combination of servers working in tandem. ‘395/3:10-14; Ex. 5 at ¶ 57. For example, at least in one embodiment disclosed in Figure 1, server 104 comprises a hardware-based server comprising high CPU throughput capabilities, large amounts of RAM for handling a large number of users, and thread pools for thread sharing. ‘395/3:14-23; Ex. 5 at ¶ 57. Because a multi-service network would necessarily require communication with a large number of users (for example, users from each of the supported messaging services), the multiple servers set up, including various role-specific sub-servers, would further assist in handling such large amounts of communication, including over the minimal bandwidth at the time. ‘395/3:10-32; Ex. 5 at ¶

57.

49. By providing server 104 as the intermediary between clients 114 and IM servers 106, server 104 acts as a carrier of the data from users to the IM network 108 and vice versa. ‘395/3:33-44; Ex. 5 at ¶ 58. This, in turn requires the unconventional handling of multiple different proprietary messaging services by server 104, including handling communications under each messaging service’s respective protocols and formats. ‘395/3:64-4:12; Ex. 5 at ¶ 58. Furthermore where the network connection to the client is potentially somewhat unreliable and lower bandwidth say over a wireless network the logging into the multiple services at the web server represents an architectural improvement over doing it local to the client. Ex. 5 at ¶ 58. In this way the added communications overhead of logging into many networks is done on a far more reliable network resulting in much less likelihood a user would get logged off of a service because multiple are running on the device and furthermore a diminished amount of traffic is transferred onto the client as needed. Ex. 5 at ¶ 58. Thus, the server-based management of multiple logins results in a smoother and better user experience. Ex. 5 at ¶ 58.

50. On the other hand, as noted herein, the convention at the time of the ‘395 and ‘453 patented inventions was a single messaging service provider handling only its respective messaging service. Ex. 5 at ¶ 59. This was due to, for example, conventional messaging service providers seeking to only invest resources in managing their own service and seeking to grow their service by requiring users to join their service to communicate with others on the service. Ex. 5 at ¶ 59. Additionally, conventional messaging service providers sought to avoid the hassle of having to manage fixes and changes to multiple messaging services and were primarily interested in having users use or adopt their proprietary system. Ex. 5 at ¶ 59. This could result in users having to maintain separate contact lists for each separate messaging service they used and could become quite cumbersome and time consuming as the user used more messaging services.

Ex. 5 at ¶ 59.

51. Another issue created by the use of multiple, often proprietary messaging services was incompatibility of contact lists between the services themselves and also incompatibility or accessibility of the contact list files on certain devices. Ex. 5 at ¶ 60. For example, at the time of the ‘395 and ‘453 inventions, it would have been conventional to copy a contact list, including an aggregated contact list file, from multiple messaging services, between devices of a similar type. Ex. 5 at ¶ 60. If one tried to do it between different types of devices such as a desktop app and a mobile device it would have been unconventional and if it was possible it would have been a cumbersome and technically involved process. Ex. 5 at ¶ 60. As a result, a user may have still been required to manually copy contact lists or manually enter individual contacts (including entering such contacts multiple times if the user was using multiple messaging services) where copying of a list was not possible, between separate different types of devices. Ex. 5 at ¶ 60. On the other hand, the claimed inventions provide an architectural solution wherein a single server obtains the user’s contact lists from multiple instant messaging services to present an aggregated list of all of the user’s contacts accessible from that server via, for example, a web browser. Ex. 5 at ¶ 60. Thus the architecture proposed by the patents was an improvement over the conventional similar device application solution. Ex. 5 at ¶ 60.

52. Similar to the embodiment of Figure 1 of the ‘395 and ‘453 Patents, the embodiment of Figure 2 likewise comprises a centralized server, server 206, which is in communication with multiple clients 214, 202, and 210, of multiple different IM networks 212, 204, and 208. ‘395/5:1-7; Ex. 5 at ¶ 61. As with the above embodiment, server 206 provides the ability to read and interpret protocols and formatting of one IM network and make that available to clients on a different IM network. ‘395/5:8-36; Ex. 5 at ¶ 61. Due to this, the ‘395 and ‘453 patented inventions provide for communications between different client types, such as desktops and

mobile devices, which may not have a specific application for any given messaging service and/or may not be able to install such applications. ‘395/5:37-49; Ex. 5 at ¶ 61.

53. In these configurations, the centralized nature of the server permits for an aggregated contact list to be maintained and displayed. ‘395/5:50-63; 3; 6:14-44; Ex. 5 at ¶ 62. Specifically, as shown in the embodiment of Figure 3, server 304 comprises network login engine 310-N for each supported messaging service (*i.e.*, low level network 302-N) which stores the login information of the user for each of said low level networks for connecting to these low level networks. ‘395/6:1-13; Ex. 5 at ¶ 62. Thus, including via network login engine 310-N, server 304 may log into each of the user’s low level networks to permit instant messaging communications, as well as access by server 304 to the user’s contact data associated with each such network. ‘395/6:13-19; Ex. 5 at ¶ 62.

54. Each of the user’s low level network contact data is aggregated into an aggregated contact list, which may be further aggregated with a contact list for the messaging service of server 304, if one is provided. ‘395/6:20-34; Ex. 5 at ¶ 63. Specifically, ‘395 and ‘453 Figures 5 and 6, and the associated specification language, disclose flowcharts depicting the methods for aggregation of the user’s contacts. ‘395/Figs. 5 and 6; 8:10-9:61; Ex. 5 at ¶ 63. As a result, including as noted herein, the user is able to see all of the user’s contacts from all of the user’s joined messaging services by server 304 providing the aggregated contact data to the user’s device for display. ‘395/Figs. 4A-4B; 6:34-7:9; & 7:41-46; Ex. 5 at ¶ 63.

55. Another unconventional and inventive aspect of the claimed inventions of the ‘395 and ‘453 Patents includes the use of a remote user profile database for instant messaging login onto multiple instant messaging networks (for example, ‘395/Claim 4; Fig. 3; & 6:30-34; Ex. 5 at ¶ 64. The user profile permitted user-specific information (for example, information used by the network login engine to login or facilitate login to the one or more networks), to be stored at the

network or server to allow contact information to be retrieved and an up-to-date aggregate contact list can be maintained, without requiring the user to enter such information each time they use the system. Ex. 5 at ¶ 64. For example, the system may maintain user login information for at least each low-level network which may then be used to, for example, by the network login engine to login or facilitate login to one or more networks, including one or more low-level networks. Ex. 5 at ¶ 64. This allows, for example, the user to provide login information once, and the system to login on behalf of the user at the server without further user action to maintain an up-to-date aggregate contact list, including by the system accessing the low-level network using the stored information and retrieving the user's contact list for that network to update the user's high level network contact list. Ex. 5 at ¶ 64. In 2006, the convention was to not store such information at the server for the server to use upon a single login, including due to security limitations and privacy concerns relating to the notion of storing instant messaging client passwords on a server to be used by the server instead of on a local device. Ex. 5 at ¶ 64. The convention was localized storage of such information to allow login to multiple servers from the local device. Ex. 5 at ¶ 64.

***2. The '395 and '453 Patents are not Directed to an Abstract Idea.***

56. The claims of the '395 and '453 Patents neither describe nor claim a concept nor a generic method or computerized system. Ex. 5 at ¶ 65. Instead, the '395 and '453 claims address, among other things, a persistent problem with messaging systems at the time of the invention whereby aggregation of contacts from a plurality of messaging services was unavailable and/or was cumbersome (for example, manually copying or transferring contact files from one messaging service (where such copying or transfer was even possible) to another messaging service); prone to potential errors (for example, manually inputting contacts could result in input errors); an inability to maintain a contact list, including an aggregated contact list, across or

between multiple user devices and subject to compatibility problems (for example, because contact lists were typically maintained in a proprietary format from messaging service to messaging service, a contact list from one messaging service was not usable or accessible by another messaging service). Ex. 5 at ¶ 65. The patented inventions enable a substantial improvement in messaging systems, including their functionality and utility. Ex. 5 at ¶ 65.

57. Further, the convention at the time was to use only a single messaging service due to not only limited computing and network resources but also because, as noted above, one could not switch between applications on a mobile phone. Ex. 5 at ¶ 66. The specific claimed inventions of the '395 and '453 Patents disclose unconventional systems and methods which solve these limitations, including, without limitation, by providing a server and web application capable of handling all inbound and outbound connections from the multiple messaging services. Ex. 5 at ¶ 66.

58. The claimed invention required substantial resources on the server side to run. Ex. 5 at ¶ 67. At the time, a single instant messaging system was far cheaper to maintain, including because every time you added a messaging system to the server was required to maintain an additional connection for each user on that additionally added system to each of the other systems to maintain a presence with the other systems. Ex. 5 at ¶ 67. IM systems had many of millions of users, so every time you added an instant messaging system to be handled by the network (from N to N+1) you decreased the potential users of an IP address, which was a very expensive proposition at the time. Ex. 5 at ¶ 67. It was far less expensive resource wise to handle this on a local client where one needed to only maintain N+1 connections. Ex. 5 at ¶ 67. With desktop applications with reliable connectivity to the network this was the conventional solutions at the time. Ex. 5 at ¶ 67.

59. As a result, the convention was singular messaging services due to the noted resource

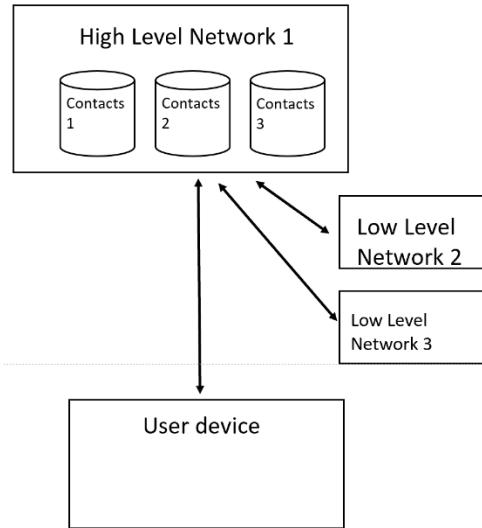
limitations requiring messaging service providers to focus on a single network, rather than multiple. Ex. 5 at ¶ 68.

60. The claims of the ‘395 and ‘453 Patents claim unconventional systems and methods which provide a network server to connect to multiple separate, proprietary messaging services to provide, including via a web application, both an aggregated contact list for all of the user’s messaging services and communications between users of all of the supported messaging services. Ex. 5 at ¶ 69. At the time, this was unconventional, including because it required significantly more resources at the network server to communicate with and manage the increased network traffic resulting from multiple users using multiple messaging services (as opposed to a single user using multiple messaging services locally). Ex. 5 at ¶ 69.

61. The patented inventions of the ‘395 Patent provide systems for contact aggregation involving a first network associated with a first messaging service provider and a second network associated with a second messaging service provider that specify, for example, “a contact aggregation engine coupled to the network login engine and the network contacts database,” which specifies the mechanism by which contacts from the first and second messaging services are accessed (*e.g.*, the network login engine) and aggregated, including that, as noted herein, the network login engine uses the user’s login information for each first and second network to which the user is a member to login or facilitate login to each such network, wherein this login information may be stored at the network or web server to allow contact information to be retrieved and an up-to-date aggregate contact list can be maintained, without requiring the user to enter such information each time they use the system. Ex. 5 at ¶ 70. Including as noted herein, this, in turn, permits the user to login and access multiple of the user’s contact lists from a central web or network application. ‘395/Figs. 5 & 6; 6:1-19; 6:34-7:9; & 8:10-9:61; Ex. 5 at ¶ 70. The particular system further specifies “a web server coupled to the network contacts database” that,

for example, allows aggregations of contacts at the network and makes the aggregated contact list available from the network and which specifies that the server provides a web application wherein, by logging into, or otherwise using, the web application, to access the user's various first and second messaging service networks, the user's contact data from the first and second messaging services is aggregated into an aggregated contact list, which may be further aggregated with a contact list for the messaging service of the server. '395/Claims 1 & 2; Ex. 5 at ¶ 70. Including as noted herein, this, in turn, provides a server-side web application which is device-independent for the user to access all of the user's contact information regardless of source network. '395/Fig. 3; 5:50-63; & 6:34-7:9. Ex. 5 at ¶ 70.

62. These limitations disclose a particular architecture and way in which contact aggregation can be accomplished from a first network associated with a first messaging service provider and a second network associated with a second messaging service, including the specific way contact aggregation is accomplished, including to provide an aggregated contact list that is stored at the network (for example, the web server) and is therefore available via the network, including via a web-based application or browser – as opposed to using conventional methods to aggregate contacts such as those described in the '395 Patent and herein. Ex. 5 at ¶ 71. An example of such architecture and functionality is as follows:



Ex. 5 at ¶ 71.

63. The patented inventions of the ‘453 Patent provide systems for contact aggregation involving a plurality of low level networks associated with a plurality of messaging service providers that specify, for example, “a contact aggregation engine coupled to the network login engine and the network contacts database,” which specifies the mechanism by which contacts from the plurality of low level networks are accessed (for example, the network login engine) and aggregated, including that, as noted herein, the network login engine uses the user’s login information for each plurality of low level networks to which the user is a member to login or facilitate login to each such network, wherein this login information may be stored at the network or server to allow contact information to be retrieved and an up-to-date aggregate contact list can be maintained, without requiring the user to enter such information each time they use the system. Ex. 5 at ¶ 72. Including as noted herein, this, in turn, permits the user to login and access multiple of the user’s contact lists from a central web or network application. ‘453/Figs. 5 & 6; 6:5-23; 6:38-7:13; & 8:18-10:4; Ex. 5 at ¶ 72. The particular system further specifies “a server coupled to the network contacts database” that, for example, allows aggregations of contacts at the network and makes the aggregated contact list available from the

network and which specifies that the server provides a server application wherein, by logging into, or otherwise using, the server application, to access the user's various low level messaging service networks, the user's contact data from the low level networks is aggregated into an aggregated contact list, which may be further aggregated with a contact list for the messaging service of the server. '453/Claims 1 & 2. Ex. 5 at ¶ 72. Including as noted herein, this, in turn, provides a server-side web application which is device-independent for the user to access all of the user's contact information regardless of source network. '453/Fig. 3; 5:52-60; & 6:38-7:13; Ex. 5 at ¶ 72.

64. These limitations disclose a particular architecture and way in which contact aggregation can be accomplished from a plurality of low level networks associated with a plurality of messaging services, including the specific way contact aggregation is accomplished, including to provide an aggregated contact list that is stored at the network (for example, the server) and is therefore available via the network, including via a server-based application or a browser – as opposed to using conventional methods to aggregate contacts such as those described in the '453 Patent and herein. Ex. 5 at ¶ 73. An example of such architecture and functionality is set forth above with respect to the '395 Patent. Ex. 5 at ¶ 73.

65. Similarly, patented inventions of the '395 Patent and '453 Patents provide methods for contact aggregation involving a high-level network and a first low level network associated with a first messaging service provider and a second low level network associated with a second messaging service provider, that specifies for example, "obtaining a first contact list associated with the first messaging service provider," "obtaining a second contact list associated with the second messaging service provider" and "maintaining an aggregated contact [list] that comprises the first contact list and the second contact list". '395/Claim 7; Ex. 5 at ¶ 74. The methods further provide for contact aggregation involving a high-level network and a plurality of low level

networks, that specifies for example, “obtaining a first contact list associated with the plurality of low level networks,” “maintaining a second contact list associated with the high level network,” “maintaining a contact list associated with the plurality of low level networks for the plurality of messaging services,” and “displaying contacts from the plurality of low level networks and the high level network in an aggregated contact list, the contacts retrieved by logging into the high level network.” ‘353/Claim 7; Ex. 5 at ¶ 74. These limitations disclose a particular way in which contact aggregation can be accomplished from a first network associated with a first messaging service provider and a second network associated with a second messaging service or from the plurality of low level networks, including the specific way contact aggregation is accomplished including to provide an aggregated contact list that comprises the first contact list and the second contact list (or the high level network and the plurality of low level networks) and may be, for example, made available via the high level network and is therefore available via the network, including via a web-based and/or server-based application or a browser – as opposed to using conventional methods to aggregate contacts such as those described in the ‘395 and ‘453 Patents and herein. Ex. 5 at ¶ 74.

- a. The ‘395 and ‘453 Claims are Directed to Innovative Computer- and Network-Based Systems and Methods.

66. None of the elements that comprise the claimed system or that are described in the claims of the ‘395 and ‘453 Patents are abstract. Ex. 5 at ¶ 75. Including as described herein and in the ‘395 and ‘453 Patents, the computer, contact aggregation engine, network login engine, web server, server and interfaces are physical or tangible things known to a person of ordinary skill in the art (“POSITA”) in light of the specification; and in view of the technological solutions and unConventionality noted herein. ‘395/3:10-32; Ex. 5 at ¶ 75.

67. As exemplified by claim 1, the subject claims of the ‘395 Patent are directed to:

1. A system comprising:

a network login engine;  
a network contacts database embodied in one or more non-transitory computer readable mediums;  
a web server coupled to the network contacts database;  
a contact aggregation engine coupled to the network login engine and the network contacts database;  
wherein, in operation, the contact aggregation engine:  
    controls the network login engine to login or facilitate login to a first network associated with a first messaging service provider and a second network associated with a second messaging service provider,  
    updates the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider,  
    maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider,  
    stores the aggregated contact list in a non-transitory computer readable medium at the web server, and  
    provides the aggregated contact list to a display device.

‘395/Claim 1; Ex. 5 at ¶ 76.

68. As exemplified by claim 7, the subject claims of the ‘395 Patent are directed to:

7. A method comprising:  
    joining a high level network;  
    joining a first low level network associated with a first messaging service provider and a second low level network associated with a second messaging service provider;  
    obtaining a first contact list associated with the first messaging service provider;  
    obtaining a second contact list associated with the second messaging service provider;  
    maintaining an aggregated contact that comprises the first contact list and the second contact list;  
    logging into the high level network;  
    displaying the aggregated contact list.

‘395/Claim 7; Ex. 5 at ¶ 77.

69. As exemplified by claim 1, the subject claims of the ‘453 Patent are directed to:

1. system for contact list aggregation across a plurality of different networks comprising:  
    a network interface;  
    a network login engine coupled to the network interface;

a network contacts database embodied in one or more non-transitory computer-readable mediums;  
a server coupled to the network contacts database;  
a contact aggregation engine coupled to the network login engine and the network contacts database;  
wherein, in operation, the contact aggregation engine controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of messaging services through a high level network using the network interface to access contact information from the plurality of messaging services, updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list, stores the aggregated contact list in a non-transitory computer-readable medium at the server, and provides the aggregated contact list including the contact information to a display device.

‘453/Claim 1; Ex. 5 at ¶ 78.

70. As exemplified by claim 7, the subject claims of the ‘453 Patent are directed to:

7. A method for contact list aggregation across a plurality of different networks comprising:
  - joining a high level network;
  - joining a plurality of low level networks associated with a plurality of messaging services through the high level network;
  - obtaining a first contact list associated with the plurality of low level networks;
  - maintaining a second contact list associated with the high level network maintaining a contact list associated with the plurality of low level networks for the plurality of messaging services;
  - logging into the high level network;
  - displaying contacts from the plurality of low level networks and the high level network in an aggregated contact list, the contacts retrieved by logging into the high level network.

‘453/Claim 7; Ex. 5 at ¶ 79.

71. As exemplified by claim 12, the subject claims of the ‘453 Patent are directed to:

12. A non-transitory computer readable medium comprising executable instructions, the instructions being executable by a processor to perform a method for contact list aggregation across a plurality of different networks, the method comprising:
  - joining a high level network;
  - joining a plurality of low level networks associated with a plurality of messaging services through the high level network;
  - obtaining a first contact list associated with the plurality of low level networks;

maintaining a second contact list associated with the high level network;  
maintaining a contact list associated with the plurality of low level networks  
for the plurality of messaging services;  
logging into the high level network;  
displaying contacts from the plurality of low level networks and the high  
level network in an aggregated contact list, the contacts retrieved by  
logging into the high level network.

‘453/Claim 12; Ex. 5 at ¶ 80.

72. Claim 1 of the ‘395 Patent, quoted above, is exemplary. Ex. 5 at ¶ 81. A POSITA would understand that the language of the ‘395 and ‘453 claims is not directed merely to a method of generically or conventionally aggregating contacts. Ex. 5 at ¶ 81. Rather, it comprises the specific aspects noted herein which provided the noted inventive, technological solutions to the problems faced by the inventors. Ex. 5 at ¶ 81. Specifically, as noted herein, the claimed inventions provide inventive, unconventional, and technological solutions to the conventional problems of accessing multiple contacts lists from multiple messaging services in a server-based web application which facilitates the user’s access to multiple networks and an aggregated contact list of all of the user’s contacts from such networks. Ex. 5 at ¶ 81. None of the elements that comprise the claimed device are abstract, as all of the computer, network contacts database, contact aggregation engine, network login engine, web server and network interfaces (‘395/Figures 1-3 and 7 (and associated description in the specification)) are physical or tangible things known to a POSITA in light of the specification; and in view of the technological solutions and unconventionality noted herein. Ex. 5 at ¶ 81.

b. *The ‘395 and ‘453 Claimed Inventions Could not be Done Manually or in One’s Head.*

73. A POSITA would understand that the claimed solutions could not be done manually, including because they necessarily require implementation via a specialized, or specially programmed, computer, including one or more networks, a contacts database, a web server, and, further, including at least login or facilitating login to a first network...and a second network

(‘395/Claim 1; Figs. 3 & 5-6; 2:11-15; 5:50-6:13; & 8:10-9:17); and joining a high level network, joining a first low level network ... and a second low level network and logging into the high level network (‘395/claim 7; Fig. 3; 5:64-6:13; & 6:35-44), nor can they be performed in a person’s head. Ex. 5 at ¶ 82. Furthermore, for example, the constant open connectivity required for receiving presence information between the server and each of the connected messaging services is not something that could be done manually or in one’s head. Ex. 5 at ¶ 82.

74. A POSITA would understand that the claimed solutions could not be done manually, including because they necessarily require implementation via a specialized, or specially programmed, computer, including one or more networks, a contacts database, a server, and, further, including at least login or facilitating login to plurality of low level networks associated with a plurality of messaging services through a high level network using the network interface to access contact information from the plurality of messaging services; updating the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list (‘453/Claims 1 and 12; Figs. 3 & 5-6; 2:5-10; 5:52-6:17); and joining a high level network, joining a plurality of low level networks associated with a plurality of messaging services through the high level network and logging into the high level network (‘453/Claim 7. 3; 6:1-17; & 6:38-47), nor can they be performed in a person’s head. Ex. 5 at ¶ 83. Furthermore, for example, the constant open connectivity required for receiving presence information between the server and each of the connected messaging services is not something that could be done manually. Ex. 5 at ¶ 83.

***3. The ‘395 and ‘453 Claimed Inventions Provide Innovative, Unconventional Concepts and Technological Solutions.***

a. *The ‘395 and ‘453 Claimed Inventions Provide Technological Solutions to Technological Problems.*

75. The technical problems addressed by the claimed inventions of ‘395 and ‘453 Patents

include aggregation of contacts from a plurality of messaging services, which, at the time of the ‘395 and ‘453 patented inventions, was difficult or cumbersome, including because, as noted herein, aggregation of contacts from a plurality of messaging services at that time was unavailable and/or was cumbersome (for example, requiring manually copying or transferring contact files from one messaging service (where such copying or transfer was even possible) to another messaging service); prone to potential errors (for example, manually inputting contacts could result in input errors); an inability to maintain a contact list, including an aggregated contact list or consistent contact list, across or between multiple user devices and subject to compatibility problems (for example, because contact lists were typically maintained in a proprietary format from messaging service to messaging service, a contact list from one messaging service was not usable or accessible by another messaging service), including via a network and/or in conjunction with the use of a web browser. Ex. 5 at ¶ 84.

76. Technical solutions provided by the claimed inventions of the ‘395 and ‘453 Patents to the technical problems faced include permitting or facilitating logging in, including from a high-level network, to first, second, or plurality of networks associated with a first, second, and/or plurality of low-level messaging services, including through one or more network login engines; acquiring, including through or with a contact aggregation engine, contact information from the first, second and/or plurality of low level networks associated with the first, second and/or plurality of messaging services and aggregating the contacts associated with the first, second and/or plurality of low level networks associated with the first, second and/or plurality of messaging services with the contact from and/or associated with the high level network and storing contacts in a network contacts database which is coupled to a web server. ‘395/Figs. 3 & 5-6; 5:50-6:12; 6:34-7:9; 8:10-41; & 9:18-61; Ex. 5 at ¶ 85. Including as noted herein, this, in turn, provides a web application which is device-independent for the user to access all of the

user's contact information regardless of source network and access and communicate with the user's contacts from each of the user's messaging services. '395/Figs. 3 & 5-6; 5:50-6:12; 6:34-7:9; 8:10-41; & 9:18-61; Ex. 5 at ¶ 85.

77. The inventions claimed in the '395 and '453 Patents further represent specific improvements in the functionality and capabilities of computer networking, databases, and messaging services and networks, including in regard to instant messaging services, systems, and network databases, including a network contacts database. Ex. 5 at ¶ 86. The inventions claimed in the '395 and '453 Patents, for example, improve the functionality of contact database systems, for example, by facilitating or allowing contacts from a first, second and/ or plurality of networks, including low level networks, (for example, messaging services or systems) to be aggregated into an aggregated contact list maintained or stored in the networks contact database. '395/1:61-2:15; Ex. 5 at ¶ 86. Including as described in the '395 and '453 Patents, and as noted herein, the claimed inventions include unconventional and inventive technological solutions to the technical problems that existed at the time, including to increasing and/or improving, for example, ease-of-use, accuracy, and efficiency in contact acquisition and management of contact databases. Ex. 5 at ¶ 86. For example, the claimed inventions of the '395 and '453 Patents, including as described herein, provide technical solutions that improve computer and database technology, including for aggregating contacts across a plurality of networks, including by aggregating a user's contacts through the use of a single network-based application and/or web service. Ex. 5 at ¶ 86. In this way, the claimed inventions of the '395 and '453 Patents reduce the use of the user's computing device and resources thereon, including the use of the user's network traffic – which, as noted herein, was highly limited at the time of the patented inventions, especially on mobile networks – because the user is not required to run multiple applications, including multiple local applications, or access multiple web services simultaneously. Ex. 5 at ¶

86. As noted herein, specifically with respect to mobile devices, it was often not possible to run multiple messaging clients at the same time and, even where it was, it would require a significant amount of bandwidth, which was often not available. Ex. 5 at ¶ 86. The inventions of the ‘395 and ‘453 Patents provide a technical solution to this problem by requiring only a single connection. Ex. 5 at ¶ 86.

b. The ‘395 and ‘453 Claimed Inventions Provide Unconventional Solutions.

78. Including as noted herein, what was conventional at the time comprised, for example, entering contact information manually for each of the user’s contacts, manually copying a contact file from one messaging service to another. Ex. 5 at ¶ 87. Another conventional way of maintaining contacts was manually copying the user’s contacts file from one device to another device used by the user which made it difficult or laborious to maintain consistent aggregated contacts lists between devices. Ex. 5 at ¶ 87.

79. These improvements include permitting users to use a singular messaging system which provides the user’s contacts from all of the user’s other messaging systems, which enables this over network or web-based systems, meaning the user’s contacts are available in their aggregated form in any location and from any device that the user desires and to furthermore organize them better as a whole. Ex. 5 at ¶ 88. By leveraging network and web-based systems, the patented invention improved upon the conventional methods of locally stored contact lists and/or proprietary contact lists which could not be accessed by systems other than the specific system that created the contact list. Ex. 5 at ¶ 88.

80. Unconventional solutions provided by the claimed inventions of the ‘395 Patent include at least logging in to and/or facilitating logging in to a first, second, or plurality of low level messaging networks from a centrally-located high level network via a web application to retrieve, via a contact aggregation engine, the user’s contact information from each of said

networks and store them in an aggregated list for access by the user via the web application regardless of where the user was located or what device the user operated to access the aggregated contact list. Ex. 5 at ¶ 89.

81. The ‘395 and ‘453 patented inventions further provided unconventional solutions by permitting the login and/or facilitating the login to multiple of the user’s messaging services via server-based web application so that the user was able to access all of the user’s messaging services via the single, device-independent web application. Ex. 5 at ¶ 90. Including as noted herein, it was unconventional to have a server-based and/or network-based application-neutral content application, as opposed to the conventional local application-specific, proprietary contacts lists that – if the contacts could even be ported or aggregated at all – had to be manually aggregated via a file exported from one local application and then imported into another local application and may not even be cross-platform transferrable. Ex. 5 at ¶ 90. Similarly, it was unconventional to have a web-based or server-based application-neutral content application, for the reasons noted above. Ex. 5 at ¶ 90.

82. Further, the asserted claims of the ‘395 and ‘453 Patents claim unconventional systems and methods which provide a centralized server to connect to multiple separate, proprietary messaging services to provide, for example via a web application, both an aggregated contact list for all of the user’s messaging services and maintaining or storing the aggregated contact list in a network contacts database accessible via the web server or server. Ex. 5 at ¶ 91.

c. The ‘395 and ‘453 Claimed Inventions Provide Substantial Benefits.

83. The claimed contact aggregation systems and methods of the ‘395 and ‘453 Patents provide the benefits of aggregating a user’s contacts from a network/web server from a plurality of networks associated with a plurality of messaging services and maintaining the aggregated contact list at a network contacts database, coupled to a web server. ‘395/Claim 1; Ex. 5 at ¶ 92.

The claimed contact aggregation systems and methods of the ‘395 Patent also provide the benefits of aggregating a user’s contacts from a high-level network from a plurality of low-level networks and maintaining the aggregated contact list at a network contacts database, coupled to a web server. ‘395/claim 7; Ex. 5 at ¶ 92. This allows, for example, a user to access their aggregated contacts from different devices. Ex. 5 at ¶ 92.

84. In addition, conventional systems and methods required manual updating, including manually copying or transferring contacts or contact lists or contact files between different user devices. Ex. 5 at ¶ 93. The claimed inventions of the ‘395 and ‘453 Patents allow, for example, contacts to be aggregated in a content-neutral and screen space saving manner accessible across a number of different devices, as contrasted with state of the art application-specific contact lists. Ex. 5 at ¶ 93. It allowed devices with lower bandwidth and intermittent internet connectivity, such as wireless or mobile devices, to work reliably with simultaneously connected instant messaging systems because the connectivity to the instant messaging systems was done on a reliably connected server. Ex. 5 at ¶ 93.

d. *The ‘395 and ‘453 Claimed Inventions Provide Inventive Solutions*

85. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and contact aggregation, and further in consideration of the ‘395 and ‘453 Patent specifications, the prosecution history and cited prior art, a POSITA would understand that the claimed “network login engine...network contacts database, web server and contact aggregations engine [for] control[ling] the network login engine to login or facilitate login to a first...and a second network associated with a first...and second messaging service provider[s to] update[] the networks contacts database with contact information obtained from [each] messaging service provider; maintain[] an aggregated contact list that comprises a...contact list associated with the contact information from [each of the] messaging service

provider[s]; [and] stor[ing] the aggregated contact list...at the web server,” including based on the use of a centralized server and device-independent web application, and including in combination with the claims of the ‘395 Patent, as a whole, is an inventive technological solution, including in view of the benefits and unconventional solutions this involves and contributes to. ‘395/Claim 1; Figs. 3 & 5-6; 1:61-2:15; 5:50-6:44; & 8:10-9:61; Ex. 5 at ¶ 94.

86. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and contact aggregation, and further in consideration of the ‘395 and ‘453 Patent specifications, the prosecution history and cited prior art, a POSITA would understand that the claimed “joining a high level network...joining a first low level network associated with a first messaging service provider and a second low level network associated with a second messaging service provider...obtaining a first contact list associated with the first messaging service provider; obtaining a second contact list associated with the second messaging service provider; maintaining an aggregated contact that comprises the first contact list and the second contact list” including based on the use of a centralized server and device-independent web application, and including in combination with the claims of the ‘395 Patent, as a whole, is an inventive technological solution, including in view of the benefits and unconventional solutions this involves and contributes to. ‘395/Claim 1; Figs. 3 & 5-6; 1:61-2:15; 5:50-6:44; & 8:10-9:61; Ex. 5 at ¶ 95.

87. For example, using certain technology claimed in the ‘395 and ‘453 Patents, (for example, the network login engine and the contact aggregation engine in conjunction with the network contacts database and web server or server), it becomes possible to, among other things, add the contacts from each of the user’s messaging networks or services to a aggregated contact list maintained at or by another network or service which the user may access from a device-neutral web application. ‘395/Figs. 3 & 5-6; 5:50-6:12; 6:34-7:9; 8:10-41; & 9:18-61; Ex. 5 at ¶

96.

88. The ‘395 and ‘453 claimed inventions comprise inventive improvements over prior technologies in order to overcome problems, including those technical problems noted herein, related to computer networks and database management (for example, related to messaging service providers) including in combination with the aggregation of contacts from multiple messaging service providers, and ability to access and communicate with the user’s contacts from each such messaging service providers, including via a network which is accessible via a web browser or similar functionality. Ex. 5 at ¶ 97. For example, the claimed inventions provide inventive solutions related to the conventional issues and inefficiencies (for example, as described herein) that were related to logging into and/or facilitating the login to a plurality of low-level networks (for example, messaging service providers) from a high-level network to obtain and aggregate contacts between such networks into an aggregated contact list that is stored or maintained in a networks contact database and which may be accessible via a network (for example, the Internet) such as through a web server. Ex. 5 at ¶ 97.

89. The ‘395 and ‘453 patented inventions further provide inventive improvements in IM network architecture, including because the unconventional multiserver-server-client architecture and shifting of communications and workload to a centralized server improve over the conventional, proprietary singular server-client architecture. Ex. 5 at ¶ 98. As noted herein, unlike conventional systems which only comprised a single server for the specific instant messaging service to connect to each client, the patented inventions utilize a multiserver-server-client architecture. Ex. 5 at ¶ 98. Including as noted herein, in so doing, the claimed inventions reduce the workload of clients and the servers of each individual instant messaging service by providing for various tasks to be run and take place on the centralized intermediary server. Ex. 5 at ¶ 98. Specifically, a POSITA would understand that the imposition of the web server between

the various instant messaging servers and/or between these instant messaging servers and each user client is inventive as an improvement in IM network architecture. Ex. 5 at ¶ 98.

### C. The ‘135 and ‘179 Patents

#### 1. *Overview of the ‘135 and ‘179 Patents*

90. The ‘135 Patent is entitled “Event Notification System And Method.” Application No. 13/165,709, filed June 21, 2011. The ‘709 Application issued on July 24, 2012 as U. S. Patent No. 8,230,135, which is a divisional of non-provisional Application No. 11/637,514, filed December 11, 2006. The ‘135 Patent also claims priority to Provisional Application No. 60/748,988, filed on December 9, 2005.

91. The ‘179 Patent shares a common title with the ‘135 Patent – “Event Notification System and Method.” Application No. 13/554,996, filed on July 20, 2012, and which issued on March 19, 2013 as U. S. Patent No. 8,402,179 Patent is a continuation of the ‘135 Patent. The ‘135 Patent and the ‘179 Patent share an essentially identical specification.

92. As the ‘179 Patent is a continuation of the application for the ‘135 Patent, the specification of the ‘179 Patent is virtually identical to the shared specification of the ‘135 Patent. Thus, for ease of reference, most specification references herein to either the ‘135 or ‘179 Patents will apply equally to the other.

#### a. *Overview of Prosecution of the ‘135 and ‘179 Patents*

93. During prosecution of the ‘135 Patent, on November 11, 2010, the patent examiner rejected then pending claims 1-20 under 35 U. S. C. § 101 as being directed to non-patentable subject matter and under 35 U. S. C. § 103(a) as being unpatentable over U. S. Published Application No. 2004/0015547 (“Griffin”) in view of U. S. Published Application No. 2001/0026231 (“Satoh”).

94. On March 12, 2012, the applicant amended the claims and argued that these claims

were directed to patentable subject matter and that the cited combination of Griffin and Satoh did not render the claims, as amended, unpatentable.

95. With regard to the rejection under 35 U. S. C. § 101, applicants noted their response to the patent examiner's rejection that claim 1 was amended "to recite, in relevant part, 'providing ...at least one of [a] plurality of character strings in [a] title array to a process executed by a processor.'" Similarly, the applicants pointed out that claim 11 was amended "to recite in relevant part, 'a means for providing ... at least one of [a] plurality of character strings in [a] title array to a process executed by a processor.'"

96. With regard to the rejection under 35 U. S. C. § 103(a), applicants pointed out that neither Griffin nor Satoh, either alone or in combination, taught, suggested or rendered obvious "'associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device;...providing an alternative title based on the at least one of the plurality of character strings to the process; [and] using the alternative title as a tile in association with the process,' as recited in [amended] claim 1." The applicants further noted that conventional Griffin did not teach, suggest or render obvious "'means for associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device; ... means for providing an alternative title based on the at least one of the plurality of character strings to the process; [and] means for using the alternative title as a title in association with the process,' as recited in [amended] claim 11. "

97. On May 24, 2012, the patent examiner issued a notice of allowance stating that "Examiner notes that applicant's arguments filed on 3/12/2012 are persuasive...."

98. The '179 Patent, which was filed on July 20, 2012, is a continuation of the '135

Patent. The examiner did not reject any of the as-filed claims and, on January 11, 2013, the examiner issued a notice of allowance.

b. *Overview of the Unconventional ‘135 and ‘179 Patented Inventions and the Conventional Technology at the Time*

99. In December 2006, options for informing or alerting a user of a messaging system, including an instant messaging system, of an event or message were limited, restricted and/or intrusive. Ex. 5 at ¶ 108. For example, when a user received a new event notification (for example, that the user had received a new message), the user was conventionally alerted by the system generating a noise, opening a pop-up window or using highlighted or blinking text. ‘135/5:30-38; Ex. 5 at ¶ 108. In addition, certain devices, such as mobile devices, had limited notification capabilities for notifying a user of an event or message. ‘135/5:20-38; Ex. 5 at ¶ 108. Generating noises, pop-up windows or using text effects, such as blinking, could be distracting or intrusive for the user. Ex. 5 at ¶ 108.

100. In addition, these conventional methods for providing event notifications (for example, noises and distracting text effects), could provide only a limited amount of information about a particular event. Ex. 5 at ¶ 109. For example, a sound could notify the user of an event but would not provide any details about the event (for example, whether there was a single event or multiple events, who a message was from, etc.). Ex. 5 at ¶ 109. Because of the limited information conveyed by conventional notification capabilities, a user was required to switch to a different application or window to determine the specific nature and/or details of the event (such as determining the number of new messages or the identity of the person sending the message), thereby interrupting the work or task of the user. Ex. 5 at ¶ 109. Further, pop-up windows (to the extent the user’s system or device was even capable of using or configured to use a pop-up window) and flashing text could be distracting and/or intrusive to the user and may only have a limited duration. Ex. 5 at ¶ 109. Lastly over time multiple events could arrive over a short period

of time that would require informing a user. Ex. 5 at ¶ 109. Changing the message immediately only when the event arrived, limited the chance to inform the user of the new event. Ex. 5 at ¶ 109.

101. Including text in the titlebar or taskbar was known. Ex. 5 at ¶ 110. However, in December 2006, the text included in the titlebar was limited to a single string of characters, for example, to identify the state of the browser or application (for example, the website the user was accessing or the name of the application). ‘135/7:3-11; Ex. 5 at ¶ 110. Certain applications could also indicate, for example, that a new message had been received through the use of an icon in the titlebar. ‘135 File History (March 12, 2012 response); Ex. 5 at ¶ 110. However, including as described by the *Griffin* reference cited by the patent examiner during prosecution of the ‘135 Patent, such applications were limited to a single line of text and used only a single icon to provide an event notification. ‘135 File History (March 12, 2012 response); Ex. 5 at ¶ 110.

102. Further, in December 2006, different user devices provided different functionality, including related to event notification and also imposed different types of restrictions for event notification. ‘135/5:20-27; Ex. 5 at ¶ 111. For example, a desktop computer may not have been capable of producing a sound for notification. Ex. 5 at ¶ 111. Certain devices and applications on those devices had inferior capabilities, including the inability to open a pop-up windows or flashing windows. ‘135/5:54-63; Ex. 5 at ¶ 111. Even where such functionality was available, it could require technical or complicated configuration changes (for example, to the browser), for which the user may not have the ability or permissions to accomplish. Ex. 5 at ¶ 111.

103. The user’s environment could also impose limitations on the type of acceptable event notifications. Ex. 5 at ¶ 112. For example, a user may not want distracting noises or pop-up windows while using certain applications or at work. ‘135/5:30-32; Ex. 5 at ¶ 112. A user that

receives many messages may not want frequently repeated noises or pop-up windows or other distracting notifications. ‘135/5:34-36; Ex. 5 at ¶ 112.

104. Providing notifications serially and/or rotating through event notifications by sending and resending events over time from a server – as opposed to including event notifications in an array that comprises multiple notifications (for example, a character string comprising a title and a character string comprising an alternative title) – results in increased bandwidth usage and usage of system resources because the same notification is sent multiple times and that the server has to keep track of what is currently being displayed in the titlebar so it knows when to send these redundant change notices. Ex. 5 at ¶ 113. The claimed inventions of the ‘135 and ‘179 Patents improve the functionality, visual effectiveness and efficiency of event notification systems as described herein by eliminating the need to send and resend notifications including by providing for the use of an array as described in the ‘135 Patent, ‘179 Patent and herein. Ex. 5 at ¶ 113.

105. The “SUMMARY” section of the ‘135 and ‘179 Patents states, in part, as follows:

A technique for user notification involves modifying a title associated with a process to include information about an event that calls for user notification. A method according to the technique may include running a process, such as, by way of example but not limitation, an IM client process, a browser, or some other process that has a title associated therewith. The method may further include processing an event, such as by way of example but not limitation, a new mail event, a new instant message event, a reminder event, a calendar event, or some other event, and generating a string of characters that includes information associated with the event. The method may further include displaying the string of characters as a title associated with the process. A system constructed according to the technique may include a client, a title array, an event processing engine, and a title provisioning engine.

‘135/1:62-2:9; Ex. 5 at ¶ 114.

106. With reference to the embodiment shown in Figure 7 (which shows a flowchart of an example of a method for displaying programmed text titles; ‘135/2:28-29), in module 702 an event that calls for user notification is processed. ‘135/9:18-19; Ex. 5 at ¶ 115.

107. At module 704, a first string of characters associated with the event is generated. ‘135/9:23-24; Ex. 5 at ¶ 116. At this point, there may or may not be a second, third, etc. string of characters associated with the event generated, as well. ‘135/9:24-26; Ex. 5 at ¶ 116.

108. At module 706, the first string of characters is stored in an array, which may also include other strings of characters associated with other events, or the first string of characters (and second, third, etc. string of characters) associated with the event may replace all current characters strings of the array. ‘135/9:27-32; Ex. 5 at ¶ 117.

109. At module 708, the first string of characters is provided from the array to a process that may include an IM client operating in a Windows® environment. ‘135/9:33-37; Ex. 5 at ¶ 118. Alternatively, the first string of characters could be provided to replace the title of a browser with the title associated with the event. ‘135/1:64-67; 7:17-23; Ex. 5 at ¶ 118.

110. At module 710, the first string of characters is used as a title in association with the process. ‘135/9:38-40; Ex. 5 at ¶ 119.

111. At module 712, a second string of characters is provided from the array to the process. ‘135/9:50-51; Ex. 5 at ¶ 120. The second string of characters could be associated with the same event as the first string of characters, an earlier event (where the second string remains in the array), or a later event (where the second string replaces or is in addition to strings associated with the last processed event). ‘135/9:51-56; Ex. 5 at ¶ 120. The second string of characters could also be unassociated with an event. ‘135/9:56-57; Ex. 5 at ¶ 120.

112. At module 714, the second string of characters is used as a title in association with the process. ‘135/9:61-62; Ex. 5 at ¶ 121.

113. With reference to the embodiment shown at Figure 5, at 502, a browser is opened. ‘135/7:1-2; Ex. 5 at ¶ 122. Next, at 504, a title associated with the state of the browser is displayed in association with the browser. ‘135/7:3-4; Ex. 5 at ¶ 122. The state of the browser

may include, for example, a current site for which the browser is displaying a page, as is shown in Figures 3A and 3C. ‘135/7:5-7; Ex. 5 at ¶ 122.

114. At module 506, an event trigger is received. ‘135/7:12-13; Ex. 5 at ¶ 123. The event trigger is associated with an event that calls for notifying a user. ‘135/7:13-15; Ex. 5 at ¶ 123. For example, the event trigger may include receipt of a new instant message. ‘135/7:15-16; Ex. 5 at ¶ 123.

115. At module 508, a title associated with the event is displayed in association with the browser. ‘135/7:17-19; Ex. 5 at ¶ 124. Thus, the title associated with the state of the browser is replaced by the title associated with the event. ‘135/7:19-21; Ex. 5 at ¶ 124.

116. Next, at module 510, it is determined whether to close the browser. ‘135/7:24-26; Ex. 5 at ¶ 125. If the browser is not to be closed (510-N), then the flowchart 500 continues to decision point 512 where it is determined whether to continue to display the title associated with the event. ‘135/7:26-29; Ex. 5 at ¶ 125. If so (512-Y), then the flowchart 500 loops back to module 508, as described previously. ‘135/7:29-30; Ex. 5 at ¶ 125. If not (512-N), then the flowchart 500 loops back to module 504, as described previously. ‘135/7:30-32; Ex. 5 at ¶ 125.

117. With reference to the embodiment shown at Figure 6, an event received on the interface 612 (from either the input device 602 or the network 604) is provided to the event processing engine 614 for processing. ‘135/8:4-6; Ex. 5 at ¶ 126.

118. Advantageously, data associated with the event can also (or in the alternative) be provided to the title provisioning engine 616. ‘135/8:11-13; Ex. 5 at ¶ 127. The title provisioning engine 616 inputs title strings, which are strings of characters, to the title array 618. ‘135/8:14-16; Ex. 5 at ¶ 127. The title array 618 includes N-1 title array strings, embodied in a computer-readable medium, which are referenced (for illustrative purposes) as title array string [0] to title array string [N]. ‘135/8:16-19; Ex. 5 at ¶ 127.

119. A given event may cause the title provisioning engine 616 to rewrite the entire title array 618 with title strings associated with the most recent events. ‘135/8:20-22; Ex. 5 at ¶ 128. Alternatively, a given event may cause the title provisioning engine 616 to append title strings to the end of the title array 618. ‘135/8:22-24; Ex. 5 at ¶ 128. In one embodiment, the title provisioning engine 616 updates the current title array 618 by adding new data associated with an event, excluding redundant data (for example, data that is already represented in the title array 618). ‘135/8:24-28; Ex. 5 at ¶ 128.

120. The multiplexer 620 selects one of the title array strings for provisioning to the client 622. ‘135/8:29-30; Ex. 5 at ¶ 129. The multiplexer 620 receives a control signal from the title provisioning engine 616 to accomplish this task. ‘135/8:30-32; Ex. 5 at ¶ 129.

121. Including as noted above, the patent specification teaches specifically how the technology improvement of the event notification systems and methods of the ‘135 and ‘179 Patents is achieved. Ex. 5 at ¶ 130. Among other innovations, the invention is able to provide to a process (such as a browser), via a title array (for example, 706), a character string associated with an event (for example, 704) for display in a titlebar or taskbar and an alternative title (for example, 712, 714) for display in a taskbar or titlebar. Ex. 5 at ¶ 130. Further, the disclosed algorithms do not display a title in the titlebar or taskbar that identifies the state of the browser or application but rather provide character strings related to a received event. Ex. 5 at ¶ 130. Thus, the invention details how the improved event notification systems and methods can be realized and how its functionality can be accomplished. Ex. 5 at ¶ 130. The patent claims recite how to implement the improved event notification systems and methods. Ex. 5 at ¶ 130. Furthermore, the claims require a non-conventional and non-generic method in order to allow for the system to provide an event notification comprising a title array for provisioning for display in a titlebar or taskbar comprising a plurality of character strings, including where at least one of the character

strings provides an alternative title. Ex. 5 at ¶ 130. Thus, the patents describe an application specific order of steps for use in a system that is not a generic or conventional arrangement. Ex. 5 at ¶ 130.

122. For example, as set forth above with regard Figure 7, the steps are performed in a relative order:

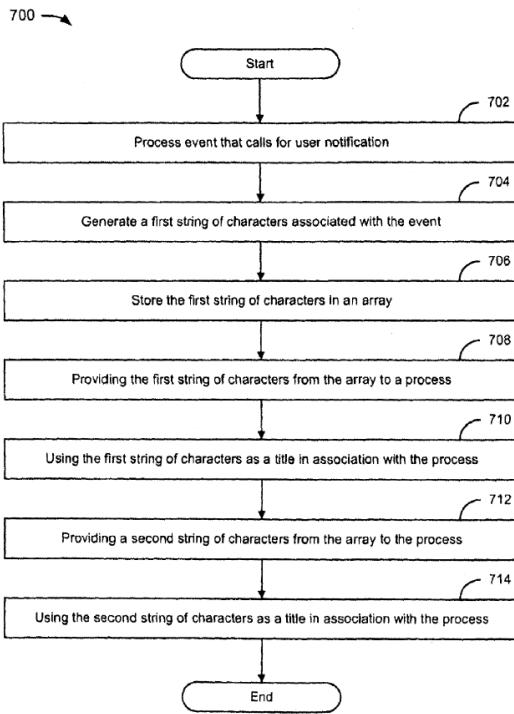


FIG. 7

Ex. 5 at ¶ 131.

123. The claimed inventions of the '135 and '179 Patents have advantages including that they allow a user to be notified of an event through use of the taskbar or titlebar thereby providing a more effective and less intrusive notification. '135/35-38; Ex. 5 at ¶ 132.

124. Another advantage of the '135 and '179 claimed inventions is that they permit the user to view notifications in the titlebar or taskbar without having to leave the window or application in which the user is working. '135/7:41-52; Ex. 5 at ¶ 133. Further, in the case of a messaging application, for example, providing an event notification in the titlebar or taskbar may

cause it to stand out from other applications the user may have open at the time. ‘135/6:13-16; Ex. 5 at ¶ 133.

125. The claimed inventions of the ‘135 and ‘179 Patents have the additional advantage that they use an array to provide a title and an alternative title for the titlebar or taskbar. ‘135/Claims 1 & 11; 8:11-28; ‘179/Claims 1 & 11; 8:15-32; Ex. 5 at ¶ 134. The use of an array to provide a title and alternative title is advantageous, including because it can provide more information to the user such as the number of messages a user has received (for example in a first title character string) and the identities of the people that sent unviewed message (for example, in a second title character string). ‘135/9:1-15; Ex. 5 at ¶ 134. Further, through the use of an array (including comprising a plurality of character strings), multiple event notifications may be presented to the user, for example, such that a prior event notification (if it remains in the array) may continue to be shown to the user even though additional notifications or information may be included as character strings in the array. Ex. 5 at ¶ 134. In this way, event notifications may be maintained (and not discarded or lost) when a new notification arrives. ‘135/8:20-28; Ex. 5 at ¶ 134. Additionally, new event notifications would not need to be maintained at the server, for example, awaiting transmission and/or display to the user or acknowledgement of the earlier notification. Ex. 5 at ¶ 134.

126. Further advantages of the claimed inventions of the ‘135 and ‘179 Patents include the use of a title array, including with a plurality of character strings and including an alternative title, can provide more information to the user including by cycling through the character strings in the title array. Ex. 5 at ¶ 135. In this way, if a user receives messages from multiple senders, for example, the identities of the senders may be included in the character strings in the array and displayed or scrolled through the taskbar or titlebar thereby providing additional information about an event or set of events. Ex. 5 at ¶ 135. Further, the title array allows previous event

notifications to remain available for display (for example, by leaving the associated character string(s) in the array) so that event notification information is not lost at the user's device when a new event notification is received. Ex. 5 at ¶ 135. In this way, the user does not have to constantly monitor the titlebar or taskbar, thus making the event notifications even less intrusive or distracting. Ex. 5 at ¶ 135. This has the further advantage that event notifications do not have to be maintained at the server. Ex. 5 at ¶ 135. Further, the claimed inventions of the '135 and '179 Patents may time multiplex the display of notifications onto the limited screen space titlebar/taskbar through the use of an array because it increases the information content related to event notifications available through that screen space over time without requiring additional resources from the server. Ex. 5 at ¶ 135.

127. An additional advantage of the '135 and '179 claimed inventions is displaying event notifications in the taskbar, including as described herein. Ex. 5 at ¶ 136. When displayed in the taskbar (including as opposed to display in a titlebar), the notifications may remain visible to the user even when the user is in another application or when the user's browser, for example, is minimized and the titlebar may not be visible. Ex. 5 at ¶ 136.

128. At the time of the '135 and '179 patented inventions, it was not known or conventional to provide an event notification in a titlebar or taskbar using an array, including to provide an alternative title. Ex. 5 at ¶ 137.

129. Although the text in a titlebar or taskbar could be changed (for example, to show the name of a website or application), the use of an array to provide both a title and an alternative title, is more efficient, including at the network/server level (for example, it would require less processing and less network bandwidth) than conventional titlebar provisioning techniques. Ex. 5 at ¶ 138.

130. The use of an array to provide a plurality of character strings for provisioning for

display in a titlebar or taskbar permits the system to provide more information to a user about an event or multiple events as opposed to the single, static titles used in conventional titlebar provisioning techniques. Ex. 5 at ¶ 139.

**2. *The ‘135 and ‘179 Claims are not Directed to an Abstract Idea.***

131. The claims of the ‘135 and ‘179 Patents neither describe nor claim a concept nor a generic method or computerized system. Ex. 5 at ¶ 140. Instead, the ‘135 and ‘179 claims address, among other things, a persistent problem with messaging systems at the time of the invention whereby event notifications were distracting, intrusive, limited by the functionality of the user’s device and/or system and did not typically or conventionally provide information related to a specific event. ‘135/5:30-38; Ex. 5 at ¶ 140. The patented inventions enable a substantial improvement in messaging systems, including their effectiveness, functionality, and utility. Ex. 5 at ¶ 140.

132. Prior to the invention, event notifications typically involved the use of intrusive and/or distracting pop-up windows or sounds, or the use of intrusive and/or distracting text effects on the titlebar (for example, blinking or flashing text) for providing event notification. ‘135/5:20-38; Ex. 5 at ¶ 141. Additionally, and as described above, these event notifications did not provide the user with very much information about the event itself or if they did pop-up with a separate notification window they were too distracting to the user, rather they were directed to simply notifying the user an event had occurred. Ex. 5 at ¶ 141.

133. The claimed inventions of the ‘135 Patent provide a particular method and system for event notification that require, for example, “associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device” which specifies how an event notification is provided and the specific data structure used to accomplish it. ‘135/Claim 1;

Ex. 5 at ¶ 142. The particular methods and systems further require “providing the at least one of the plurality of character strings in the title array to a process executed by a processor” which specifies the information that can be provided for display. ‘135/Claim 1; Ex. 5 at ¶ 142. The particular method and systems also require “providing an alternative title based on the at least one of the plurality of character strings to the process” and “using the alternative title as a title in association with the process” which further specifies additional information that may be provided for display and how such additional information is provided. ‘135/Claim 1; Ex. 5 at ¶ 142. These limitations disclose a particular way in which an event notification can be displayed, including the specific information to be displayed and the data structure used to provide the information – as opposed to using conventional methods to display event notifications such as those described in the ‘135 Patent and herein. Ex. 5 at ¶ 142.

134. The claimed inventions of the ‘179 Patent provide a particular method and system for event notification that require, for example, “generating an event notification for the event” and “storing the event notification in an array” which specifies how an event notification is provided and the specific data structure used to accomplish it. ‘179/Claim 1; Ex. 5 at ¶ 143. The particular methods and systems further require “using the event notification as a title in association with the process” which specifies the information that can be provided for display. ‘179/Claim 1; Ex. 5 at ¶ 143. The particular method and systems also require “providing an alternative title from the array to the process” and “using the alternative title as a title in association with the process” which further specifies additional information that may be provided for display. ‘179/Claim 1; Ex. 5 at ¶ 143. These limitations disclose a particular way in which an event notification can be displayed, including the specific information to be displayed and the data structure used to provide the information – as opposed to using conventional methods to display event notifications such as those described in the ‘179 Patent and herein. Ex. 5 at ¶ 143.

a. The ‘135 and ‘179 Claims are Directed to Innovative Computer- and Network-Based Systems and Methods.

135. None of the elements that comprise the claimed system or that are described in the claims method are abstract. Ex. 5 at ¶ 144. Including as described herein and in the ‘135 and ‘179 Patents, the computer, event processing engine, title provisioning engine and interfaces are physical or tangible things known to a POSITA in light of the specification; and in view of the technological solutions and unconventionality noted herein. Ex. 5 at ¶ 144.

136. As exemplified by claim 1, the subject claims of the ‘135 Patent are directed to:

1. A method comprising:

receiving information of an event that calls for user notification;  
generating an event notification for the event;  
associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device;  
providing the at least one of the plurality of character strings in the title array to a process executed by a processor;  
providing an alternative title based on the at least one of the plurality of character strings to the process;  
using the alternative title as a title in association with the process.

‘135/Claim 1; Ex. 5 at ¶ 145.

137. As exemplified by claim 11, the subject claims of the ‘135 Patent are directed to:

11. A system comprising:

a means for receiving information of an event that calls for user notification;  
a means for generating an event notification for the event;  
a means for associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device;  
a means for providing the at least one of the plurality of character strings in the title array to a process executed by a processor;  
a means for providing an alternative title based-on the at least one of the plurality of character strings to the process;  
a means for using the alternative title as a title in association with the process.

‘135/Claim 11; Ex. 5 at ¶ 146.

138. As exemplified by claim 1, the subject claims of the ‘179 Patent are directed to:

1. A method comprising:

processing an event that calls for user notification;  
generating an event notification for the event;  
storing the event notification in an array;  
providing the event notification from the array to a process executed by a processor;  
using the event notification as a title in association with the process;  
providing an alternative title from the array to the process;  
using the alternative title as a title in association with the process.

‘179/Claim 1; Ex. 5 at ¶ 147.

139. As exemplified by claim 11, the subject claims of the ‘179 Patent are directed to:

11. A system comprising:

a means for processing an event that calls for user notification;  
a means for generating an event notification for the event;  
a means for storing the event notification in an array;  
a means for providing the event notification from the array to a process;  
a means for using the event notification as a title in association with the process;  
a means for providing an alternative title from the array to the process;  
a means for using the alternative title as a title in association with the process.

‘179/Claim 11; Ex. 5 at ¶ 148.

140. Claim 1 of the ‘135 Patent, quoted above, is exemplary. Ex. 5 at ¶ 149. A POSITA would understand that the language of the ‘135 and ‘179 claims is not directed merely to a method of providing a generic or conventional event notification to a user. Ex. 5 at ¶ 149. Rather, they comprise the aspects noted herein which provided inventive, technological solutions to the problems faced by the inventors. Ex. 5 at ¶ 149. None of the elements that comprise the claimed device are abstract, as all of the computer, event processing engine, title provisioning engine and interfaces are physical or tangible things known to a POSITA in light of the specification; and in view of the technological solutions and unconventionality noted herein. Ex. 5 at ¶ 149.

b. *The ‘135 and ‘179 Claimed Inventions Could not be Done Manually or in One’s Head.*

141. A POSITA would understand that the claimed solutions could not be done manually including because they necessarily require implementation via a computer processor including to receive and process event notifications, associating the event notification with at least one of a plurality of character strings in a title array, providing character strings to a process executed by a processor, a display device including a titlebar or taskbar (‘135/Claim 1); a means for receiving information of an event that calls for user notification; a means for generating an event notification for the event; a means for associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device; a means for providing the at least one of the plurality of character strings in the title array to a process executed by a processor; a means for providing an alternative title based-on the at least one of the plurality of character strings to the process; a means for using the alternative title as a title in association with the process (‘135/Claim 11); to process an event, generate an event notification, store the event notification in an array, provide the event notification to a process executed by a processor (‘179 Patent/Claim 1); and/or a means for processing an event that calls for user notification; a means for generating an event notification for the event; a means for storing the event notification in an array; a means for providing the event notification from the array to a process; a means for using the event notification as a title in association with the process; a means for providing an alternative title from the array to the process; a means for using the alternative title as a title in association with the process (‘179/Claim 11). Ex. 5 at ¶ 150. Nor can they be performed in a person’s head. Ex. 5 at ¶ 150.

***3. The ‘135 and ‘179 Claimed Inventions Provide Innovative, Unconventional Concepts and Technological Solutions.***

***a. The ‘135 Claimed Inventions Provide Technological Solutions To Technological Problems.***

142. The technical problems addressed by the ‘135 and ‘179 Patents include those noted herein, including intrusive and/or distracting pop-up windows or sounds and the use of distracting text effects (for example, blinking or flashing windows) for providing event notification, as well as the inability of certain devices or applications to provide the certain notifications due to technical limitations, for example, no sound, or a need to configure an application or device which might be difficult for a user or beyond his or her abilities or permissions. ‘135/5:30-38; Ex. 5 at ¶ 151. Conventional event notification methods, such as those described herein, were also limited in the amount of information they could provide about an event. Ex. 5 at ¶ 151. For example, the user may be notified by a sound that an event had occurred, but would have no information about the specific nature of that event (for example, if a new message had been received or who the message was from). Ex. 5 at ¶ 151. Conventional methods further required the user to check the window and/or application to determine whether there is a relevant event (such as a message from a relevant person). ‘135/7:42-52; Ex. 5 at ¶ 151. Further, titlebar/taskbar messaging that did exist was limited in information bandwidth that it provided based on the prior art by not having a means to time multiplex the screen area the title and taskbar took up. Ex. 5 at ¶ 151.

143. Technical solutions provided by the claimed inventions of the ‘135 Patent and ‘179 Patent to technical problems faced include associating an event notification with a plurality of character strings in a title array, providing a character string from the array to the event notification process, such as an instant messaging application or browser and providing an alternative title from the title array to the event notification process. Ex. 5 at ¶ 152. Technical

solutions provided by the claimed inventions of the ‘135 and ‘179 Patents further include providing improved and more effective user notification methods that make it easier and more efficient for users to visualize and navigate through multiple notifications; providing for more efficient, uninterrupted and less bandwidth dependent handling of notifications locally by the browser; including, for example, by providing a title array comprising a plurality of character strings for display in a titlebar or taskbar of a display device and further including providing an alternative title from the title array. ‘135/5:20-38; 5/54-62; Ex. 5 at ¶ 152.

b. The ‘135 and ‘179 Claimed Inventions Provide Unconventional Solutions.

144. Including as noted herein, what was conventional at the time comprised notifying a user of an event with a pop-up window or noise, or the use of distracting flashing windows and/or text effects on the titlebar. ‘135/5:30-38; Ex. 5 at ¶ 153. It was also conventional to provide, for example, a single icon on the titlebar or taskbar to indicate an event, including as shown in the *Griffin* reference described herein. Ex. 5 at ¶ 153. For example, as described in the file history of the ‘135 Patent. Ex. 5 at ¶ 153. ‘135 File History (March 12, 2012 response). Ex. 5 at ¶ 153. However, a user at work may not want to play a noise or have a popup window show up every time a message is received. ‘135/5:31-32; Ex. 5 at ¶ 153. Further, a user may want or benefit from more subtle and information notifications, including, for example, without having to stop her other activities to close pop-up windows. Ex. 5 at ¶ 153. In addition, the user’s device may lack functionality to provide conventional event notifications, for example, sounds or pop-up windows. Ex. 5 at ¶ 153.

145. Unconventional solutions provided by the claimed inventions of the ‘135 and ‘179 Patents include at least providing an event notification in the titlebar or taskbar including through the use of an array comprising a plurality of character strings and including an alternative title. Ex. 5 at ¶ 154.

146. At the time of the ‘135 and ‘179 claimed inventions, it was not conventional to provide event notifications, including through the use of an array and including an alternative title, in the titlebar or taskbar. Ex. 5 at ¶ 155.

c. *The ‘135 and ‘179 Claimed Inventions Provide Substantial Benefits.*

147. An advantage of notifying a user of an event in a titlebar or taskbar (including as opposed to using a sound or pop-up window) is that it provides a more effective, higher information content over time, and subtle notification to the user, for example, no frequent and/or intrusive ping-sound or pop-up windows. ‘135/5:20-38; 5:54-63; Ex. 5 at ¶ 156. As set forth above, the specification of the ‘135 and ‘179 Patents states that “a user at work may not want to play a noise or have a popup window show up every time a message is received.” ‘135/5:31-32; Ex. 5 at ¶ 156.

148. The claimed inventions of the ‘135 and ‘179 Patents permit a user to receive more effective (for example, by conveying a greater amount of information) event notifications in the titlebar or taskbar without having to stop or interrupt other activities and also eliminates distracting notifications such as from noises or, for example, distracting text or visual effects such as blinking of the titlebar, taskbar or window. Ex. 5 at ¶ 157. Furthermore by time multiplexing the titlebar/taskbar space more information content can be shown over time. Ex. 5 at ¶ 157.

149. The claimed inventions of the ‘135 and ‘179 Patents provide additional information to the user, including by providing notifications in the title bar (for example, with a string of characters which may be used to indicate the number of messages a user has received and/or the identity of a person sending a message to the user) and including by providing an alternative title. Ex. 5 at ¶ 158.

150. Further, the use of an array comprising a plurality of character strings related to an

event for display in a titlebar or taskbar provides a user with more detailed information about a particular event in a less intrusive and/or distracting way. Ex. 5 at ¶ 159. For example, a character string displayed in a titlebar may include information about the number of messages a user has received and an alternative title may provide additional information about the identity of the sender of a message. Ex. 5 at ¶ 159. In this way, the user may be able to determine whether the event requires an immediate response or is of particular importance without having to stop the user's current task and/or without having to switch to another window or application to determine whether the event is of particular importance. Ex. 5 at ¶ 159.

d. The '135 and '179 Claimed Inventions Provide Inventive Solutions.

151. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of the '135 Patent specification, the prosecution history and cited prior art, a POSITA would understand that "associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device" including based on an event notification and including in combination with the other claim elements of the '135 Patent as a whole is an inventive technological solution, including in view of the benefits and unconventional solution this involves and contributes to. Ex. 5 at ¶ 160.

152. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of the '135 Patent specification, the prosecution history and cited prior art, a POSITA would understand that "providing the at least one of the plurality of character strings in the title array to a process executed by a processor" including in combination with the other claim elements of the '135 Patent as a whole is an inventive technological solution, including in view of the benefits

and unconventional solution this involves and contributes to. Ex. 5 at ¶ 161.

153. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of the ‘135 Patent specification, the prosecution history and cited prior art, a POSITA would understand that “providing an alternative title based on the at least one of the plurality of character strings to the process” including in combination with the other claim elements of the ‘135 Patent as a whole is an inventive technological solution, including in view of the benefits and unconventional solution this involves and contributes to. Ex. 5 at ¶ 162.

154. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of the ‘135 Patent specification, the prosecution history and cited prior art, a POSITA would understand that “using the alternative title as a title in association with the process” including in combination with the other claim elements of the ‘135 Patent as a whole is an inventive technological solution, including in view of the benefits and unconventional solution this involves and contributes to. Ex. 5 at ¶ 163.

155. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of the ‘179 Patent specification, the prosecution history and cited prior art, a POSITA would understand that “providing the event notification from the array to a process executed by a processor” including in combination with the other claim elements of the ‘179 Patent as a whole is an inventive technological solution, including in view of the benefits and unconventional solution this involves and contributes to. Ex. 5 at ¶ 164.

156. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of

the ‘179 Patent specification, the prosecution history and cited prior art, a POSITA would understand that “using the event notification as a title in association with the process” including in combination with the other claim elements of the ‘179 Patent as a whole is an inventive technological solution, including in view of the benefits and unconventional solution this involves and contributes to. Ex. 5 at ¶ 165.

157. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of the ‘179 Patent specification, the prosecution history and cited prior art, a POSITA would understand that “providing an alternative title from the array to the process” including in combination with the other claim elements of the ‘179 Patent as a whole is an inventive technological solution, including in view of the benefits and unconventional solution this involves and contributes to. Ex. 5 at ¶ 166.

158. Consistent with the above discussion, including the problems solved that had been faced by conventional messaging systems and event notification, and further in consideration of the ‘179 Patent specification, the prosecution history and cited prior art, a POSITA would understand that “using the alternative title as a title in association with the process” including in combination with the other claim elements of the ‘179 Patent as a whole is an inventive technological solution, including in view of the benefits and unconventional solution this involves and contributes to. Ex. 5 at ¶ 167.

159. The avoidance of intrusive or distracting pop-windows, noises or text effects was unconventional and inventive. Ex. 5 at ¶ 168. Further, the use of an array to provide a string of characters for display in a titlebar or taskbar as well as an alternative title for display in a taskbar or titlebar was inventive. Ex. 5 at ¶ 168.

**D. The Claims of the Patents-in-Suit do not Unreasonably Preempt their Respective Fields.**

160. Including as noted herein, the ‘395 Patent and ‘453 Patent do not claim merely the abstract idea of “aggregating contact lists” that provides no inventive concept. Ex. 5 at ¶ 169. Instead, the ‘395 Patent and ‘453 Patent claim specific methods and systems for contact aggregation, including from a first and second messaging system and/or by a high-level network from a plurality of low-level networks (for example, messaging services), including via a contact aggregation engine that controls a network login engine to facilitate login to the low-level networks to update the contact database with contact information from the low-level networks and wherein an aggregated contact list is maintained or stored at the network contacts database (including as described in detail herein) where infringement of the patent claims can be readily avoided while still practicing the alleged abstract idea proposed by LinkedIn, given that the patent claims do not read on the alleged abstract idea. Ex. 5 at ¶ 169. Indeed, the claims of these patents do not aggregate contacts as in the prior art but instead aggregate contacts by a high-level network from a plurality of low-level networks as discussed extensively herein. Ex. 5 at ¶ 169.

161. For example, “aggregating contact lists” may be practiced outside of the limited scope of the patent claims at least by:

- a. The use of a system such as that described in the Kaplan reference (U.S. Patent No. 7,496,379), cited by the patent examiner;
- b. The use of locally stored contacts, including those maintained for each of the user’s messaging systems;
- c. The use of proprietary and/or non-standardized contact files which can be combined by the user; or
- d. The use of a single messaging system maintained locally at the user’s device.

Ex. 5 at ¶ 170.

162. Including as noted herein, the ‘135 Patent and ‘179 Patent do not claim merely the abstract idea of “notifying a user of an event in the title of a display” that provides no inventive

concept. Ex. 5 at ¶ 171. Instead, the ‘135 Patent and ‘179 Patent claim highly specific combinations of event notification methods and systems including associating an event notification with a plurality of character strings in an array and providing an alternative title based on the plurality of characters strings (including as described in detail herein) where infringement of the patent claims can be readily avoided while still practicing the alleged abstract idea proposed by Defendants. Ex. 5 at ¶ 171.

163. For example, “notifying a user of an event in the title of a display,” may be practiced outside of the limited scope of the patent claims at least by:

- a. The use of a system such as that described in the Griffin reference (U.S. Publication No. 2004/0015547), cited by the patent examiner;
- b. The use of an audible sound or noise;
- c. The use of a pop-up window; or
- d. The use of intrusive or distracting flashing or text effects.

Ex. 5 at ¶ 172.

#### **COUNT I – INFRINGEMENT OF U.S. PATENT No. 8,510,395**

164. Plaintiff is the owner of the ‘395 Patent and it has all substantial rights to the ‘395 Patent, including the right and standing to sue and recover damages for past, present, and future infringement of the patent.

165. Claim 1 of the ‘395 Patent covers a system comprising “a network login engine; a network contacts database embodied in one or more non-transitory computer readable mediums; a web server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine: controls the network login engine to login or facilitate login to a first network associated with a first messaging service provider and a second network associated with a second messaging service provider, updates the networks contacts database with contact

information obtained from the first messaging service provider and the second messaging service provider, maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device.”

166. Claim 7 of the ‘395 Patent covers a method comprising “joining a high level network; joining a first low level network associated with a first messaging service provider and a second low level network associated with a second messaging service provider; obtaining a first contact list associated with the first messaging service provider; obtaining a second contact list associated with the second messaging service provider; maintaining an aggregated contact [list] that comprises the first contact list and the second contact list; logging into the high level network; displaying the aggregated contact list.”

167. LinkedIn has infringed, and is now infringing, the ‘395 patent, including at least claims 1 and 7, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising the practicing, without authority from Plaintiff, systems and methods for obtaining and aggregating contact information from a plurality of messaging services providers via LinkedIn’s LinkedIn Application system, including as claimed in the ‘395 asserted claims. On information and belief, LinkedIn practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising [www.linkedin.com](http://www.linkedin.com).

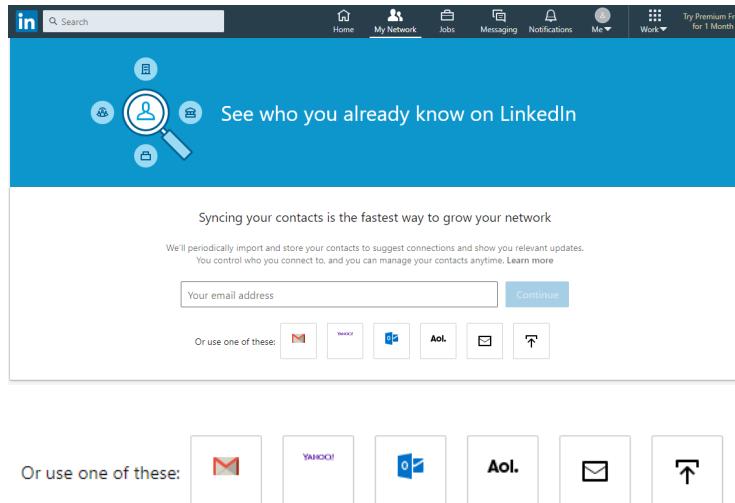
168. Without limitation, the accused system comprising the LinkedIn Application system that comprises a network login engine, a network contacts database, a web server and a contact aggregation engine, wherein the contact aggregation controls the network login engine to controls the network login engine to login or facilitate login to a first network associated with a

first messaging service provider and a second network associated with a second messaging service provider, updates the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider, maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device.

169. Without limitation and for example, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods to permit a user to login via the LinkedIn Application to a first and second messaging service provider, obtain a first contact list from the first messaging service provider and a second contact list from the second messaging service provider, and updating the network contacts.

170. Further, the LinkedIn Application comprises systems and methods which permit a user to login via the LinkedIn Application to a first and second messaging service provider, obtain a first contact list from the first messaging service provider and a second contact list from the second messaging service provider, and update the network contacts.

171. For example, the LinkedIn Application permits a user to join multiple networks associated with respective messaging service providers which interact with, *inter alia*, APIs of LinkedIn and the other networks:



See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>

A screenshot of the LinkedIn 'Manage synced sources' page. It shows two main sections: 'Contacts' and 'Calendar'.  
**Contacts:** A heading 'Manage synced sources' with a 'Remove all' button. A note says 'Syncing your contact information helps you keep in touch with your most important connections, so you always know the right times to reach out.' Below are four sync entries:

- Google (Sync button)
- Outlook - Personal (Learn how)
- Outlook - Work (Learn how)
- Phone contacts (Learn how)

  
**Calendar:** A heading 'Sync your calendar' with a 'Remove all' button. A note says 'Syncing your calendar lets you see your meeting history with people, and we'll tell you who's in your next meeting.' Below are two sync entries:

- Google (Sync button)
- Phone calendar (Learn how)

[← Back to Manage Your Contacts](#)

A screenshot of the LinkedIn 'Manage Syncing' page, showing the 'Contacts' section.  
**Contacts:** A heading 'Sync your contacts' with a 'Remove all' button. A note says 'Syncing your contact information helps you keep in touch with your most important connections, so you always know the right times to reach out.' Below is one sync entry:

- Google (Sync button)

See, e.g., LinkedIn User Manage Syncing Settings page at <https://www.linkedin.com/mynetwork/settings/manage-syncing/>

#### Contact and Calendar Information

We receive personal data (including contact information) about you when others import or sync their contacts or calendar with our Services, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

#### Syncing Contacts from Other Address Books and Sources

[LinkedIn Contacts Manager](#) can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

**Note:** If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.

To sync contacts:

1. Click the  My Network icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  Manage synced contacts near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

##### Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the  My Network icon at the top of your LinkedIn homepage.
  2. Click **Contacts** under **Manage my network** on the left rail.
  3. Click  Add more contacts on the right rail.
- Note:** You'll be redirected to a page where you can enter the source you want to import the contacts from.

**Note:** If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

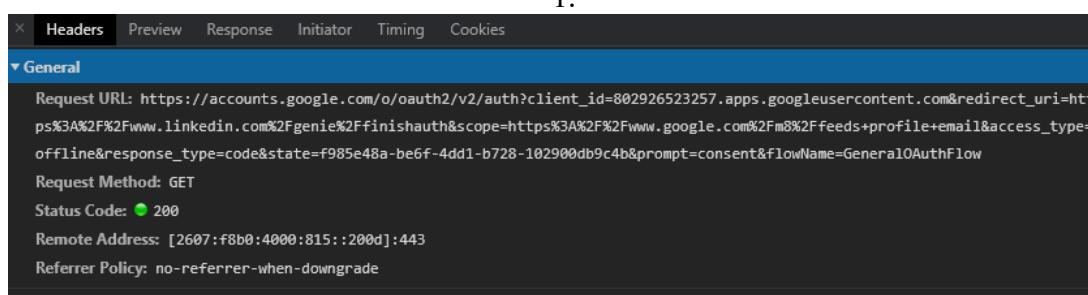
**Important:** Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>

1.



The screenshot shows a browser developer tools Network tab with the 'Headers' tab selected. The Request URL is [https://accounts.google.com/o/oauth2/v2/auth?client\\_id=802926523257.apps.googleusercontent.com&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm%8%2Ffeeds+profile+email&access\\_type=offline&response\\_type=code&state=f985e48a-be6f-4dd1-b728-102900db9c4b&prompt=consent&flowName=GeneralOAuthFlow](https://accounts.google.com/o/oauth2/v2/auth?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm%8%2Ffeeds+profile+email&access_type=offline&response_type=code&state=f985e48a-be6f-4dd1-b728-102900db9c4b&prompt=consent&flowName=GeneralOAuthFlow). The Request Method is GET, Status Code is 200, and the Remote Address is [2607:f8b0:4000:815::200d]:443. The Referrer Policy is no-referrer-when-downgrade.

See, e.g., Google Sign-In Request page at [https://accounts.google.com/o/oauth2/v2/auth/identifier?client\\_id=802926523257.apps.googleusercontent.com&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm%8%2Ffeeds%20profile%20email](https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm%8%2Ffeeds%20profile%20email)

172. For example, the LinkedIn Application obtains a contact list from each of the user's said connected networks for aggregating, maintaining, and displaying:

The screenshot shows the LinkedIn desktop interface. At the top, there is a navigation bar with the LinkedIn logo, a search bar, and links for Home, My Network, Jobs, and Messaging. Below the navigation bar, a dropdown menu shows "Saved contacts" and "Google ▾". The main content area displays a list of "3 Imported Contacts" sorted by "Recently added". Each contact entry includes a profile picture, name, company, creation date, and a "Connect" button. The first two contacts have "Connect" buttons, while the third has a "Pending" status indicator.

Profile Picture	Name	Company	Created	Status
	John Edwards	Participant in [group] (1 member)	Created: 12:24 PM	Connect
	Jason Edwards	Participant in [group] (1 member)	Created: 12:24 PM	Connect
	Jason Edwards	Participant in [group] (1 member)	Created: 12:24 PM	⌚ Pending

This screenshot shows the LinkedIn desktop interface with imported contacts from Yahoo. The layout is identical to the Google screenshot above, featuring a "Saved contacts" dropdown and a "Yahoo ▾" option. It lists three contacts from the "Recently added" group. Each contact has a "Connect" button next to it.

Profile Picture	Name	Company	Created	Status
	Jason Edwards	Participant in [group] (1 member)	Created: 12:34 PM	Connect
	Jason Edwards	Participant in [group] (1 member)	Created: 12:34 PM	Connect
	Jason Edwards	Participant in [group] (1 member)	Created: 12:34 PM	Connect

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

The screenshot shows a table titled "Manage your contacts" with a sub-instruction "These are the contacts you have imported through an address book import". The table has columns for "Source" (with icons for Google, Yahoo, and LinkedIn), "Name", and "Contact Information". There are checkboxes for selecting multiple contacts and a "Delete selected" button.

Source	Name	Contact Information
Google	[REDACTED]	[REDACTED]
Yahoo	[REDACTED]	[REDACTED]
LinkedIn	[REDACTED]	[REDACTED]
Google	[REDACTED]	[REDACTED]
LinkedIn	[REDACTED]	[REDACTED]
Google	[REDACTED]	[REDACTED]
Yahoo	[REDACTED]	[REDACTED]

See, e.g., User LinkedIn Manage Your Contacts page at  
<https://www.linkedin.com/mynetwork/contacts/>

## Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically **import and store** details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our **Privacy Policy**.

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at  
<https://www.linkedin.com/help/linkedin/answer/98247>

## LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using **LinkedIn Contacts Manager**. Collecting and storing all your contacts in one place will help to keep them up to date.

**Note:** Your **contacts and connections** aren't the same thing.

You can begin using LinkedIn Contacts Manager by **importing and inviting your email contacts**, **creating and uploading a contacts file**, and **syncing contacts from other sources**. Currently, you can only sync Google Calendar and Google Contacts, so **old sources may be missing**.

Once you've synced and imported your contacts, you can **delete specific contacts** to remove them from your **LinkedIn Contacts Manager address book**. If you've accidentally sent invitations to the contacts you imported, you can **withdraw the invitations**.

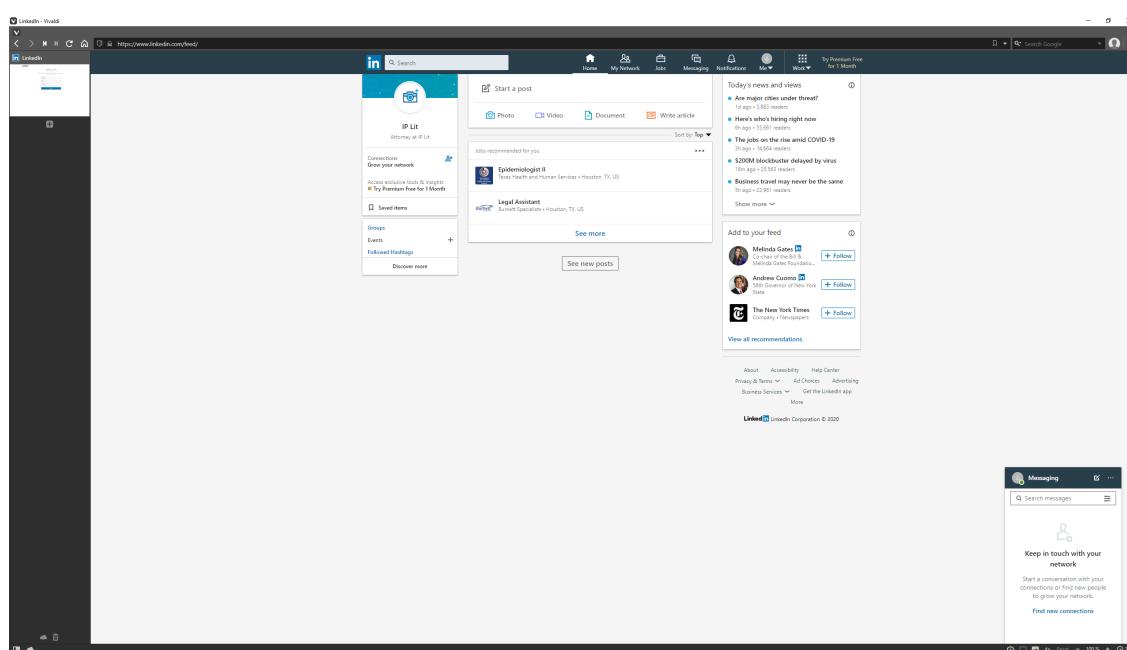
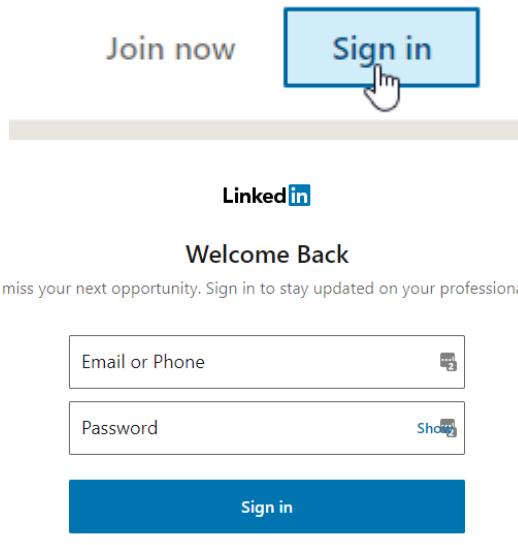
LinkedIn takes your privacy and the security of your data seriously. Learn more about the **privacy of your information in LinkedIn Contacts Manager**, and how you can **access your account data**.

Last updated: 7 months ago

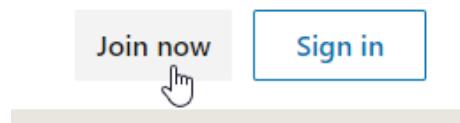
See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at  
<https://www.linkedin.com/help/linkedin/answer/91972>

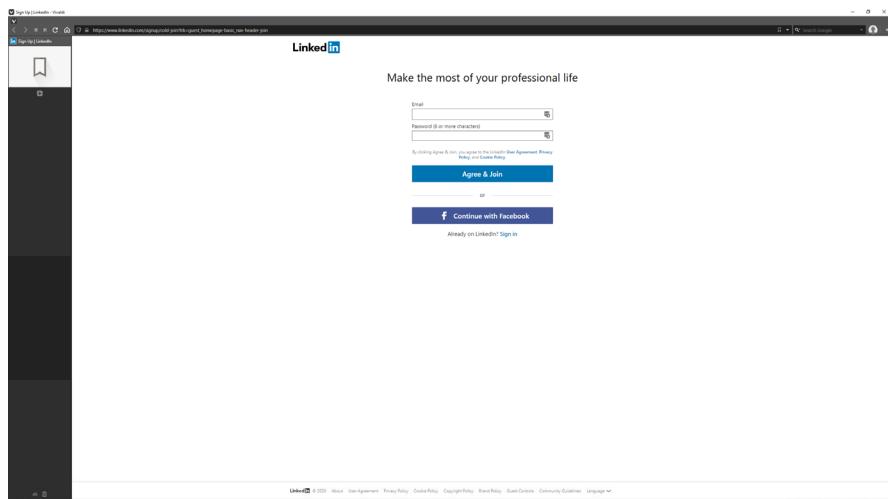
173. The LinkedIn Application comprises a network login engine. For example, an engine for use by the user to log into the user's LinkedIn account and/or the user's social media

or other networks or, if the user does not presently have a LinkedIn account, they may create a new account:



See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>





## Welcome, IP!

Let's start your profile, connect to people you know, and engage with them on topics you care about.

Country/Region \*

City/District \*

Next

Your profile helps you discover the right people and opportunities

Most recent job title \*

Most recent company \*

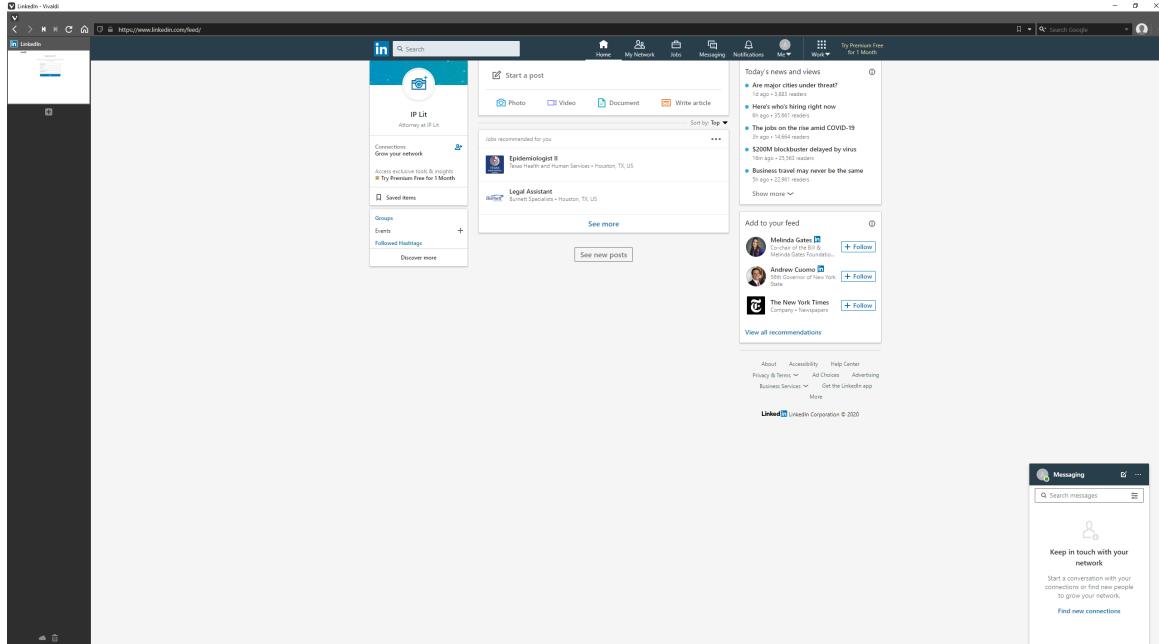
Industry \*

▼

 There are 3527937 members in the same industry on LinkedIn.

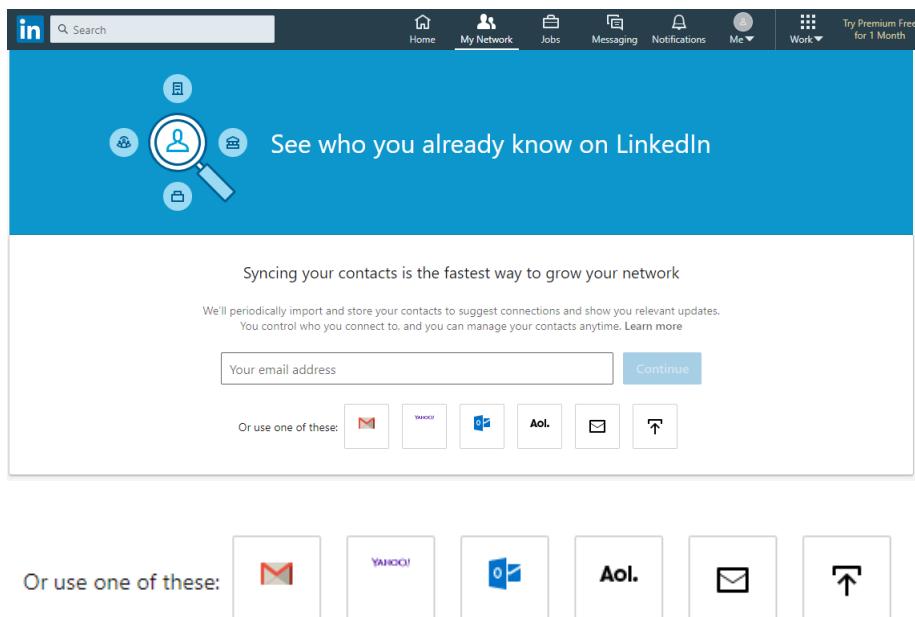
I'm a student

Continue



See, e.g., LinkedIn Signup page at <https://www.linkedin.com/signup/cold-join>

174. Further, the LinkedIn Application permits the user to login to a number of different networks, and the user may further connect said network to the user's LinkedIn account, which interacts with APIs of LinkedIn and these other networks and redirects the user to these other networks for signing in:



See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>

Manage synced sources

To learn more about our privacy and security practices, visit our [Privacy Policy](#) page.  
[Learn more in the Help Center](#)

Contacts	Remove all
Syncing your contact information helps you keep in touch with your most important connections, so you always know the right times to reach out.	
 Google	<a href="#">Sync</a>
 Outlook - Personal	<a href="#">Learn how</a>
 Outlook - Work	<a href="#">Learn how</a>
 Phone contacts	<a href="#">Learn how</a>

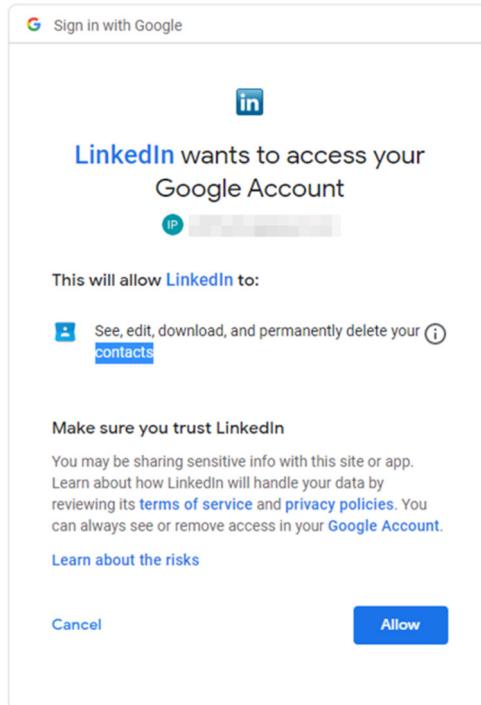
  

Calendar	Remove all
Syncing your calendar lets you see your meeting history with people, and we'll tell you who's in your next meeting.	
 Google	<a href="#">Sync</a>
 Phone calendar	<a href="#">Learn how</a>

[← Back to Manage Your Contacts](#)

See, e.g., LinkedIn User Manage Syncing page  
<https://www.linkedin.com/mynetwork/settings/manage-syncing/>

```
▼ Query String Parameters    view source    view URL encoded
client_id: 802926523257.apps.googleusercontent.com
redirect_uri: https://www.linkedin.com/genie/finishauth
scope: https://www.google.com/m8/feeds profile email
access_type: offline
response_type: code
state: f985e48a-be6f-4dd1-b728-102900db9c4b
prompt: consent
flowName: GeneralOAuthFlow
```



See, e.g., Google Sign-In Request page at [https://accounts.google.com/o/oauth2/v2/auth/identifier?client\\_id=802926523257.apps.googleusercontent.com&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email](https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email)

175. The LinkedIn application comprises a network contacts database embodied in one or more non-transitory computer readable mediums. For example, the LinkedIn Application comprises an address book comprising at least the user's LinkedIn contacts embodied in, for example, a hard drive and/or other ROM and/or, at a minimum, RAM, for storing the user's contact list in said database on LinkedIn's servers:

### LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

**Note:** Your [contacts and connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at

<https://www.linkedin.com/help/linkedin/answer/91972>

## Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically import and store details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at  
<https://www.linkedin.com/help/linkedin/answer/98247>

### Contact and Calendar Information

We receive personal data (including contact information) about you when others import or sync their contacts or calendar with our Services, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

## Syncing Contacts from Other Address Books and Sources

[LinkedIn Contacts Manager](#) can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

**Note:** If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.

To sync contacts:

1. Click the My Network icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click Manage synced contacts near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

#### Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the My Network icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click Add more contacts on the right rail.

**Note:** You'll be redirected to a page where you can enter the source you want to import the contacts from.

**Note:** If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

**Important:** Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at  
<https://www.linkedin.com/help/linkedin/answer/1278>

The screenshot shows the LinkedIn desktop website interface. At the top, there is a navigation bar with the LinkedIn logo, a search bar, and links for Home, My Network, Jobs, and Messaging. Below the navigation bar, the main content area has a header "Saved contacts" and a dropdown menu showing "Google" and "Yahoo". A tooltip indicates "3 Imported Cont". The main content area displays a list of "3 Imported Contacts" sorted by "Recently added". Each contact entry includes a profile picture, name, company, location, and creation date (e.g., "Created: 12:24 PM"). There are "Connect" buttons next to each contact.

Saved contacts **Google**

3 Imported Cont

in Search

Home My Network Jobs Messaging

Saved contacts **Google**

3 Imported Contacts

Sort by: Recently added ▾

Search by name or company

Maria Rodriguez  
Marketing Associate at LinkedIn Corp.  
Created: 12:24 PM **Connect**

John Edwards  
Marketing Associate at LinkedIn Corp.  
Created: 12:24 PM **Connect**

Anna Peterson  
Marketing Associate  
Created: 12:24 PM **Pending**

Saved contacts **Yahoo**

3 Imported Cont

in Search

Saved contacts **Yahoo**

3 Imported Contacts

Sort by: Recently added ▾

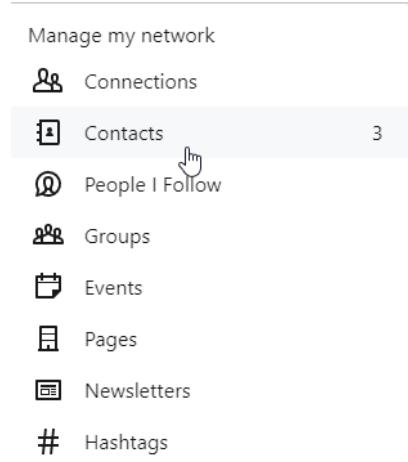
Search by name or company

Maria Rodriguez  
Marketing Associate at LinkedIn Corp.  
Created: 12:34 PM **Connect**

John Edwards  
Marketing Associate  
Created: 12:34 PM **Connect**

Anna Peterson  
Marketing Associate  
Created: 12:34 PM **Connect**

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>

Saved contacts Yahoo ▾

Google

3 Imported Cont Yahoo

Saved contacts Yahoo ▾

3 Imported Contacts Search by name or company

Sort by: Recently added ▾

Profile Picture	Name	Title	Created	Connect
	John Doe	Software Engineer	Created: 12:34 PM	Connect
	Jane Smith	Marketing Specialist	Created: 12:34 PM	Connect
	Mike Johnson	Operations Manager	Created: 12:34 PM	Connect

Headers Preview Response Initiator Timing Cookies

General

Request URL: [https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List\(YAHOO\\_CONTACTS\)&start=3](https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List(YAHOO_CONTACTS)&start=3)

Request Method: GET

Status Code: 200

Remote Address: [2620:1ec:21::14]:443

Referrer Policy: no-referrer-when-downgrade

Response Headers

```
cache-control: no-cache, no-store
content-encoding: gzip
content-length: 204
content-security-policy: default-src 'none'; style-src 'none' 'report-sample'; script-src 'none' 'report-sample';
report-uri https://www.linkedin.com/platform-telemetry/csp?f=jv
content-type: application/vnd.linkedin.normalized+json+2.1; charset=UTF-8
date: Mon, 27 Jul 2020 20:55:12 GMT
expect-ct: max-age=86400, report-uri="https://www.linkedin.com/platform-telemetry/ct"
expires: Thu, 01 Jan 1970 00:00:00 GMT
pragma: no-cache
status: 200
strict-transport-security: max-age=2592000
```

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

176. The LinkedIn Application comprises a web server coupled to the network contacts database. For example, the LinkedIn Application comprises a web server connected to the LinkedIn contact database:

The screenshot shows the LinkedIn desktop website. At the top, there is a "Welcome Back" message and a sign-in form with fields for "Email or Phone" and "Password". Below the sign-in button is a "Forgot password?" link and a "New to LinkedIn? Join now" link. The bottom half of the screen displays a browser developer tools interface, specifically the Network tab, showing a list of requests made by the browser. One request is highlighted, showing details such as the Request URL ([https://www.linkedin.com/feed/?trk=guest\\_homepage-basic\\_nav-header-signin](https://www.linkedin.com/feed/?trk=guest_homepage-basic_nav-header-signin)), Request Method (GET), Status Code (200), and Response Headers including "cache-control: no-cache, no-store", "content-encoding: gzip", and "content-security-policy: default-src 'none'; header-uri 'self'; form-action 'self'". Below the developer tools, the LinkedIn feed page is visible, showing news items, profile cards, and a messaging sidebar.

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>

177. The LinkedIn Application comprises a contact aggregation engine coupled to the network login engine and the network contacts database. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager which permits LinkedIn's servers to aggregate the user's contacts and is coupled to the LinkedIn login engine and to the storage medium comprising the user's LinkedIn contacts:

### LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

**Note:** Your [contacts and connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts, creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

### Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically [import and store](#) details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

#### Contact and Calendar Information

We receive personal data (including contact information) about you when others [import or sync their contacts or calendar with our Services](#), associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

## Syncing Contacts from Other Address Books and Sources

[LinkedIn Contacts Manager](#) can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

**Note:** If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.

To sync contacts:

1. Click the  My Network icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Manage synced contacts** near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

### Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the  My Network icon at the top of your LinkedIn homepage.
  2. Click **Contacts** under **Manage my network** on the left rail.
  3. Click  **Add more contacts** on the right rail.
- Note:** You'll be redirected to a page where you can enter the source you want to import the contacts from.

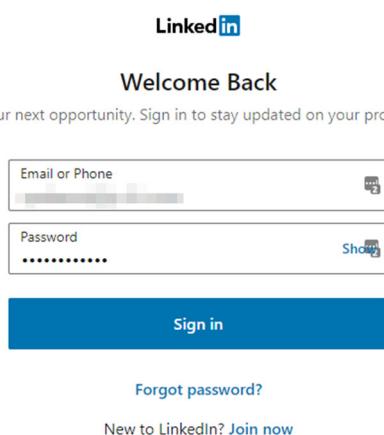
**Note:** If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

**Important:** Synced contacts are not automatically invited to connect with you on LinkedIn.

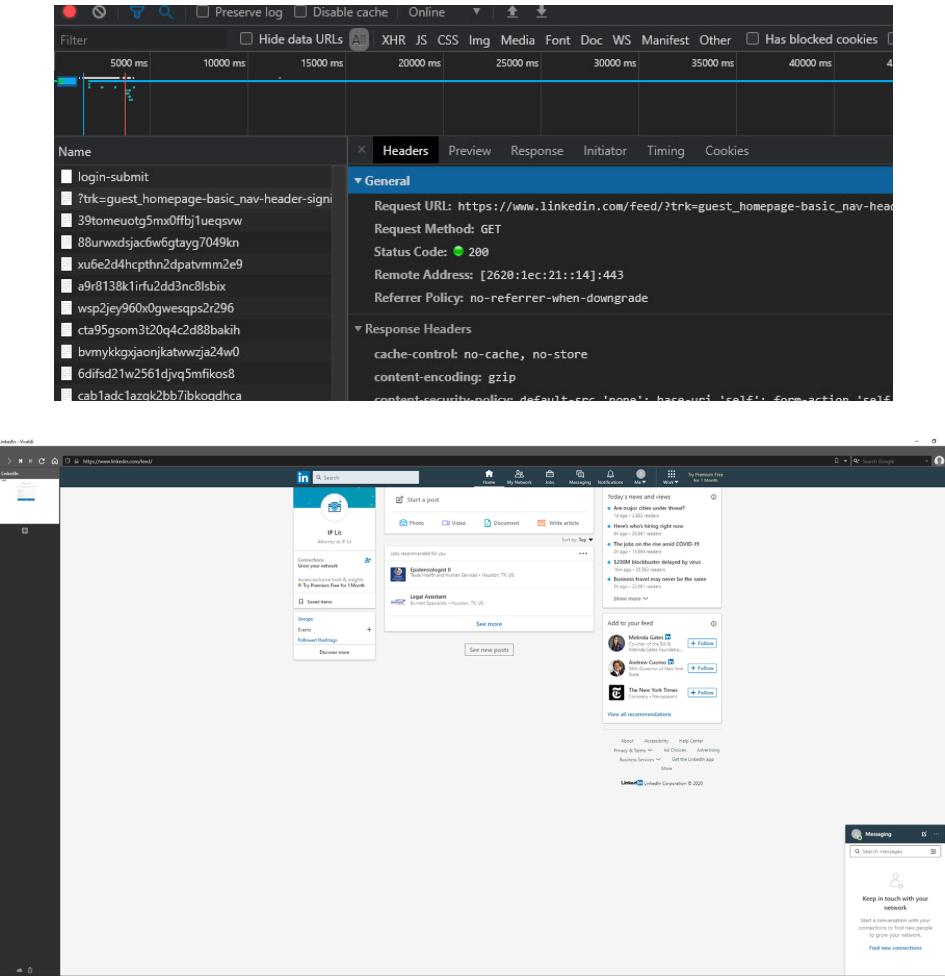
Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

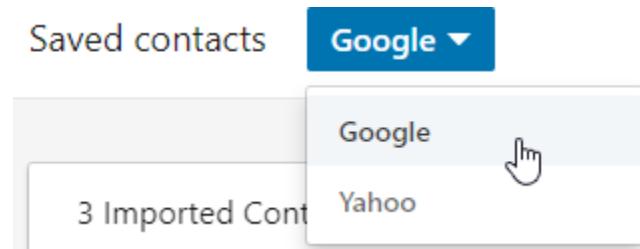
See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at  
<https://www.linkedin.com/help/linkedin/answer/1278>



The image shows the LinkedIn login page. At the top, there is a blue header bar with the LinkedIn logo. Below it, the text "Welcome Back" is displayed. A message encourages users to "Don't miss your next opportunity. Sign in to stay updated on your professional world." There are two input fields: "Email or Phone" and "Password". The "Email or Phone" field contains a blurred email address. The "Password" field contains several dots. To the right of the "Password" field is a "Show" link. Below the input fields is a large blue "Sign in" button. At the bottom of the form, there are links for "Forgot password?" and "New to LinkedIn? Join now".



See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>



This screenshot shows the LinkedIn interface with the 'Saved contacts' tab selected. A dropdown menu above the list indicates the source is 'Google'. The list displays three imported contacts, all created at 12:24 PM:

- John Edwards**  
Created: 12:24 PM [Connect](#)
- John Edwards**  
Created: 12:24 PM [Connect](#)
- John Palmer**  
Created: 12:24 PM [Pending](#)

A search bar at the top right allows users to search by name or company.

This screenshot shows the LinkedIn interface with the 'Saved contacts' tab selected. A dropdown menu above the list indicates the source is 'Yahoo'. The list displays three imported contacts, all created at 12:34 PM:

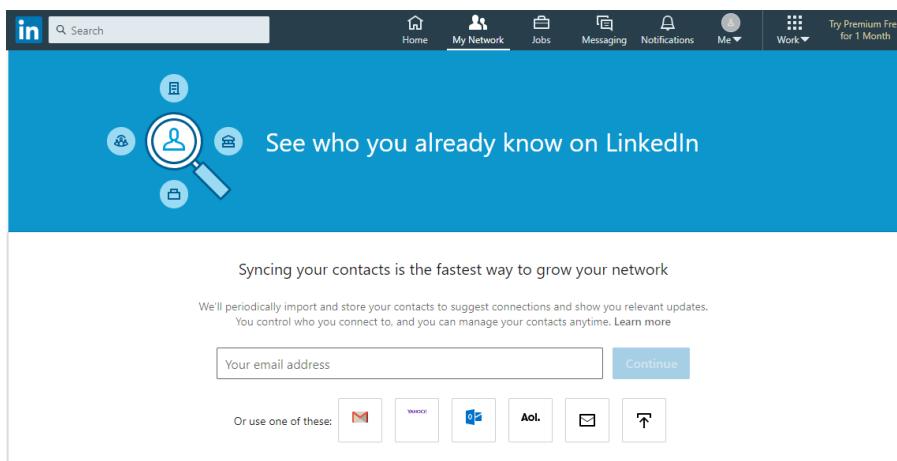
- John Edwards**  
Created: 12:34 PM [Connect](#)
- John Edwards**  
Created: 12:34 PM [Connect](#)
- John Palmer**  
Created: 12:34 PM [Connect](#)

A search bar at the top right allows users to search by name or company. A mouse cursor is hovering over the 'Yahoo' button in the dropdown menu.

The screenshot shows a network request details panel. The request URL is [https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List\(YAHOO\\_CONTACTS\)&start=3](https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List(YAHOO_CONTACTS)&start=3). The request method is GET, status code is 200, and the remote address is [2620:1ec:21::14]:443. The response headers include cache-control: no-cache, no-store, content-encoding: gzip, content-length: 204, content-security-policy: default-src 'none'; style-src 'none' 'report-sample'; script-src 'none' 'report-sample'; report-uri https://www.linkedin.com/platform-telemetry/csp?f=jv, content-type: application/vnd.linkedin.normalized+json+2.1; charset=UTF-8, date: Mon, 27 Jul 2020 20:55:12 GMT, expect-ct: max-age=86400, report-uri="https://www.linkedin.com/platform-telemetry/ct", expires: Thu, 01 Jan 1970 00:00:00 GMT, pragma: no-cache, status: 200, strict-transport-security: max-age=2592000.

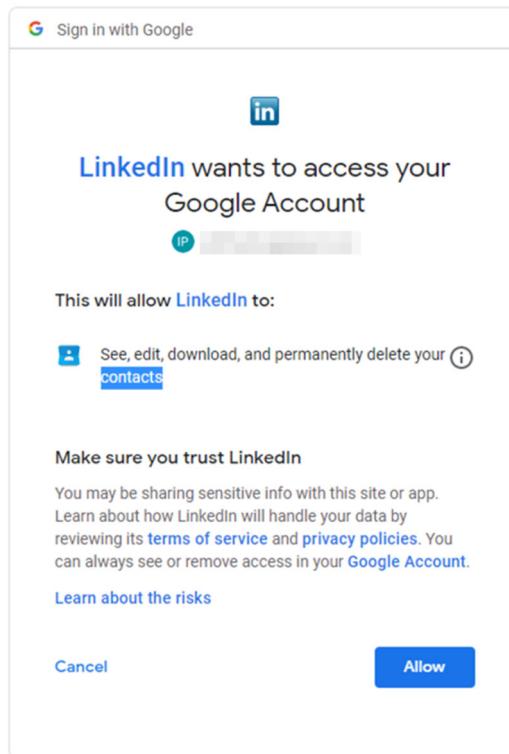
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

178. In operation, the contact aggregation engine (see above) controls the network login engine to login or facilitate login to a first network associated with a first messaging service provider and a second network associated with a second messaging service provider. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including facilitating the user logging into and associating the user's various social media and/or other networks, including via APIs of LinkedIn and/or APIs of the user's social media and/or other networks:





See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>



See, e.g., Google Sign-In Request page at  
[https://accounts.google.com/o/oauth2/v2/auth/identifier?client\\_id=802926523257.apps.googleusercontent.com&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email](https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email)

Hi, [REDACTED]

By agreeing, you'll sign in to  
Linkedin with your Yahoo account  
and allow Linkedin to access:

 Profiles  
Read and write

Read and update your extended profile information like email address, first name, last name, birthday, pictures etc.

 Yahoo Contacts  
Read

By allowing access, you're letting this application view, import and store your Yahoo Address book information.

I agree to the Yahoo OpenID and OAuth terms

 Agree       Not now

See, e.g., Yahoo! Sign-In Request page at  
[https://api.login.yahoo.com/oauth2/request\\_auth\\_fe?client\\_id=dj0yJmk9S0c3Y2RXVm1RTQxJmQ9WVdrOVIYWmxXbWxDTXpBbWNHbzlNQS0tJnM9Y29uc3VtZXJzZWNyZXQmeD02MA--&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=openid%20sdpp-w%20sdct-r&sec=true&response\\_type=code&state=5ef8e091-7714-423c-a57bf607710145fb&nonce=6612439201265914688&guccounter=1&guce\\_referrer=aHR0cHM6Ly9sb2dpbi5YWhvby5jb20v&guce\\_referrer\\_sig=AQAAADx0ox5D1DZWYJqN1T31M8wjeY72WHnm5USj7t1-994XN4dtTs9wXZPxbFnm9LuyM26LTbfNgpUgeh89NDgcnS2xsBPJoliDu6onlExDhf7tsoQEJKVkkqOu2ac4W3pj3P34o\\_G23xo8DLX4pMAqIKiP6olGZjKhodFpcMc2yR](https://api.login.yahoo.com/oauth2/request_auth_fe?client_id=dj0yJmk9S0c3Y2RXVm1RTQxJmQ9WVdrOVIYWmxXbWxDTXpBbWNHbzlNQS0tJnM9Y29uc3VtZXJzZWNyZXQmeD02MA--&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=openid%20sdpp-w%20sdct-r&sec=true&response_type=code&state=5ef8e091-7714-423c-a57bf607710145fb&nonce=6612439201265914688&guccounter=1&guce_referrer=aHR0cHM6Ly9sb2dpbi5YWhvby5jb20v&guce_referrer_sig=AQAAADx0ox5D1DZWYJqN1T31M8wjeY72WHnm5USj7t1-994XN4dtTs9wXZPxbFnm9LuyM26LTbfNgpUgeh89NDgcnS2xsBPJoliDu6onlExDhf7tsoQEJKVkkqOu2ac4W3pj3P34o_G23xo8DLX4pMAqIKiP6olGZjKhodFpcMc2yR)

179. In operation, the contact aggregation engine (*see* above) updates the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including updating the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider, including obtaining and importing the contacts of the first network onto LinkedIn's servers after

the user logs in to the first network and the contacts of the second network onto LinkedIn's servers after the user logs in to the second:

### Syncing Contacts from Other Address Books and Sources

**LinkedIn Contacts Manager** can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

**Note:** If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.

To sync contacts:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Manage synced contacts** near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

#### Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Add more contacts** on the right rail.

**Note:** You'll be redirected to a page where you can enter the source you want to import the contacts from.

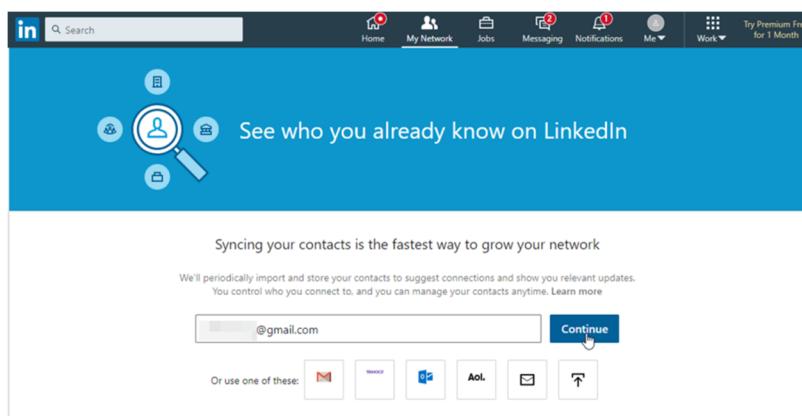
**Note:** If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

**Important:** Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>



Saved contacts Google ▾

Google

Yahoo

The screenshot shows the LinkedIn homepage with a dark header. The top navigation bar includes the LinkedIn logo, a search bar, and links for Home, My Network, Jobs, and Messaging. A notification icon in the top right corner shows 2 notifications. Below the header, a dropdown menu shows "Saved contacts" and "Google". The main content area displays a list titled "3 Imported Contacts" sorted by "Recently added". Each contact entry includes a profile picture, name, company, creation date, and a "Connect" button.

Profile Picture	Name	Company	Created	Action
	John Edwards	President of Edwards & Associates, Inc.	Created: 12:24 PM	Connect
	David Edwards	President of Edwards & Associates, Inc.	Created: 12:24 PM	Connect
	David Edwards	President of Edwards & Associates, Inc.	Created: 12:24 PM	Pending

This screenshot shows the LinkedIn sync interface. It features a blue header with the LinkedIn logo, a search bar, and navigation links for Home, My Network, Jobs, Messaging, Notifications, Me, and Work. A "Try Premium Free for 1 Month" button is also present. The main content area has a blue background with a central graphic of a person icon surrounded by icons for email, phone, and messaging. Text reads "See who you already know on LinkedIn". Below this, a message encourages syncing contacts to grow the network, stating: "Syncing your contacts is the fastest way to grow your network. We'll periodically import and store your contacts to suggest connections and show you relevant updates. You control who you connect to, and you can manage your contacts anytime. Learn more". A text input field contains "@yahoo.com" and a "Continue" button. Below the input field, there's a row of icons for other email providers: Gmail, Yahoo!, Outlook, AOL, and others.

The screenshot shows the LinkedIn homepage after importing contacts from Yahoo. The top navigation bar and "Saved contacts" dropdown are identical to the previous screen. The main content area displays a list titled "3 Imported Contacts" sorted by "Recently added". Each contact entry includes a profile picture, name, company, creation date, and a "Connect" button.

Profile Picture	Name	Company	Created	Action
	John Edwards	President of Edwards & Associates, Inc.	Created: 12:34 PM	Connect
	David Edwards	President of Edwards & Associates, Inc.	Created: 12:34 PM	Connect
	David Edwards	President of Edwards & Associates, Inc.	Created: 12:34 PM	Connect

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

180. In operation, the contact aggregation engine (*see above*) maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including maintaining an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device:

#### **LinkedIn Contacts Manager - Overview**

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

**Note:** Your [contacts](#) and [connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

## Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically import and store details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

### Contact and Calendar Information

We receive personal data (including contact information) about you when others import or sync their contacts or calendar with our Services, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

The screenshot shows the LinkedIn desktop interface. At the top, there's a navigation bar with icons for Home, My Network, Jobs, and Messaging. Below it, a search bar and a 'Saved contacts' button are visible. A 'Google' dropdown menu is open, showing 'Google' and 'Yahoo' as options, with a cursor pointing at 'Yahoo'. The main content area displays a list of '3 Imported Contacts' with a 'Sort by: Recently added' dropdown. Each contact entry includes a profile picture, name, company, creation date, and a 'Connect' button. One contact has a green 'Pending' status indicator.

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

This screenshot shows the same LinkedIn desktop interface as the previous one, but with a different 'Google' dropdown menu. It lists 'Google' and 'Yahoo' again, with a cursor pointing at 'Yahoo'. The main content area shows the same list of '3 Imported Contacts' as the previous screenshot.

The screenshot shows a 'Saved contacts' section on a LinkedIn interface. At the top, it says '3 Imported Contacts' and 'Sort by: Recently added'. Below this, there are three contact entries, each with a small profile picture, the contact's name, their title, and the text 'Created: 12:34 PM'. To the right of each entry is a blue 'Connect' button. A search bar at the top right says 'Search by name or company'.

*See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>*

181. In operation, the contact aggregation engine (*see* above) stores the aggregated contact list in a non-transitory computer readable medium at the web server. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including storing the user's aggregated contact list on LinkedIn's servers:

The screenshot shows a sidebar menu titled 'Manage my network'. It includes sections for 'Connections' (with 'Contacts' highlighted), 'People I Follow', 'Groups', 'Events', 'Pages', 'Newsletters', and 'Hashtags'. A hand cursor icon is positioned over the 'Contacts' link. The 'Connections' section has a count of '3' next to it.

*See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>*

Saved contacts **Google ▾**

3 Imported Cont 

in  Home My Network Jobs Messaging **2**

Saved contacts **Google ▾**

3 Imported Contacts  Sort by: Recently added ▾

Profile Picture	Name	Created	Action
	Michael Johnson Software engineer at LinkedIn Corp.	Created: 12:24 PM	<b>Connect</b>
	John Edwards Software engineer at LinkedIn Corp.	Created: 12:24 PM	<b>Connect</b>
	Bruce Palmer Software engineer at LinkedIn Corp.	Created: 12:24 PM	 Pending

Saved contacts **Yahoo ▾**

3 Imported Cont 

Saved contacts **Yahoo ▾**

3 Imported Contacts  Sort by: Recently added ▾

Profile Picture	Name	Created	Action
	Michael Johnson Software engineer at LinkedIn Corp.	Created: 12:34 PM	<b>Connect</b>
	Bruce Palmer Software engineer at LinkedIn Corp.	Created: 12:34 PM	<b>Connect</b>
	John Edwards Software engineer at LinkedIn Corp.	Created: 12:34 PM	<b>Connect</b>

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

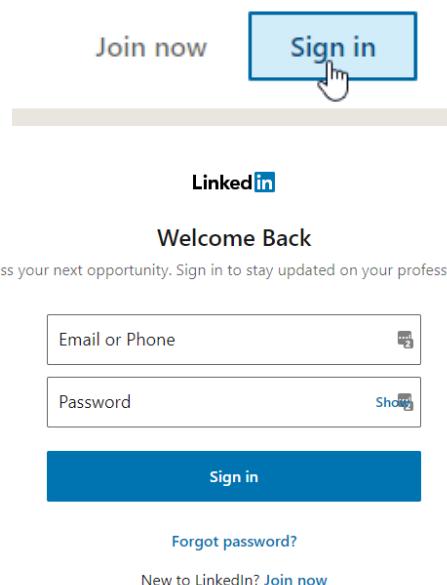
The screenshot shows a network request details panel. At the top, there are tabs for Headers, Preview, Response, Initiator, Timing, and Cookies. The Headers tab is selected. Below the tabs, under the General section, the Request URL is listed as [https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List\(YAHOO\\_CONTACTS\)&start=3](https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List(YAHOO_CONTACTS)&start=3). The Request Method is GET, and the Status Code is 200. Other details include Remote Address [2620:1ec:21::14]:443 and Referrer Policy no-referrer-when-downgrade.

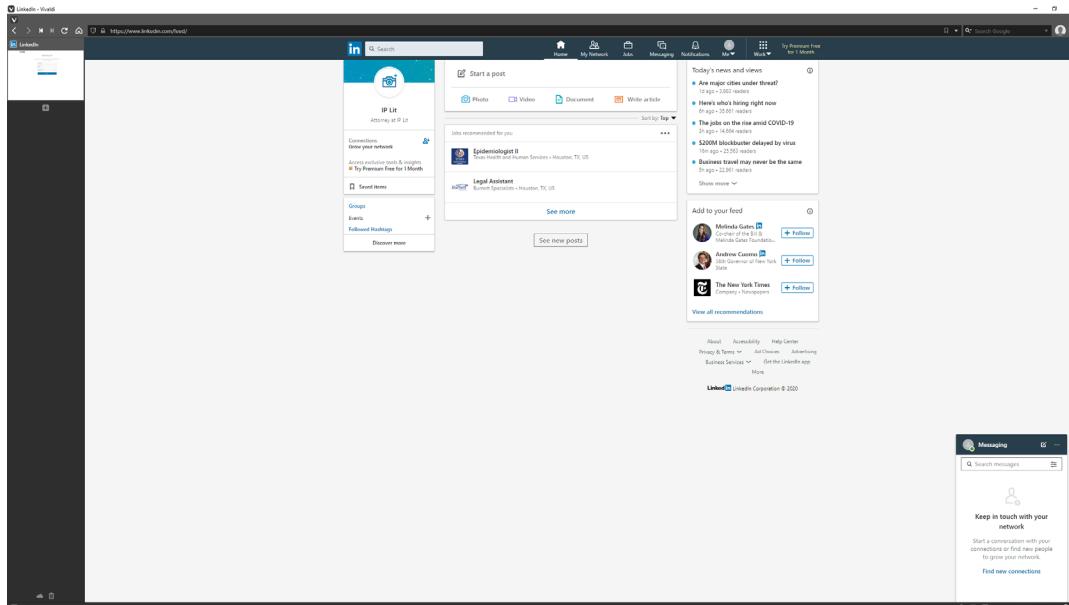
Under the Response Headers section, various HTTP headers are listed:

- cache-control: no-cache, no-store
- content-encoding: gzip
- content-length: 204
- content-security-policy: default-src 'none'; style-src 'none' 'report-sample'; script-src 'none' 'report-sample'; report-uri https://www.linkedin.com/platform-telemetry/csp?f=jv
- content-type: application/vnd.linkedin.normalized+json+2.1; charset=UTF-8
- date: Mon, 27 Jul 2020 20:55:12 GMT
- expect-ct: max-age=86400, report-uri="https://www.linkedin.com/platform-telemetry/ct"
- expires: Thu, 01 Jan 1970 00:00:00 GMT
- pragma: no-cache
- status: 200
- strict-transport-security: max-age=2592000

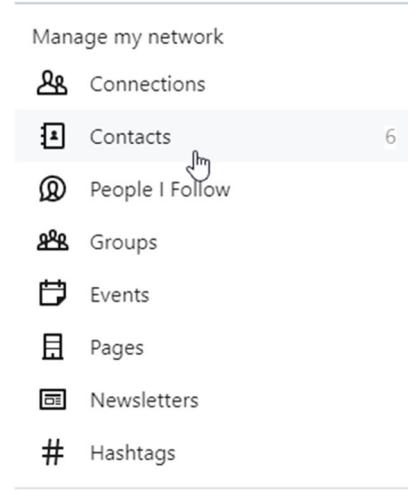
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

182. In operation, the contact aggregation engine (see above) provides the aggregated contact list to a display device. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including providing the user's aggregated contact list stored on LinkedIn's servers to the user's device for display to the user:

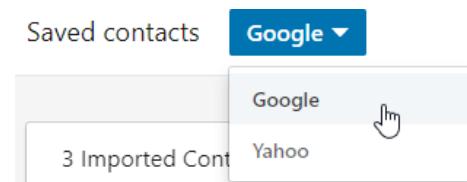




See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>



See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>



The image consists of three vertically stacked screenshots of the LinkedIn 'Saved contacts' page.

**Screenshot 1 (Top): LinkedIn - Google**

- Header: LinkedIn logo, Search bar, Home, My Network, Jobs, Messaging.
- Section: 'Saved contacts' (highlighted in blue), 'Google ▾' (highlighted in blue).
- Content: '3 Imported Contacts' (highlighted in blue). A dropdown menu shows 'Google' (highlighted in blue) and 'Yahoo' (highlighted in blue with a cursor icon).
- Table: Shows three imported contacts:
  - David Johnson**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:24 PM', 'Connect' button.
  - John Edwards**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:24 PM', 'Connect' button.
  - David Johnson**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:24 PM', 'Pending' status.

**Screenshot 2 (Middle): LinkedIn - Yahoo**

- Header: LinkedIn logo, Search bar, Home, My Network, Jobs, Messaging.
- Section: 'Saved contacts' (highlighted in blue), 'Yahoo ▾' (highlighted in blue).
- Content: '3 Imported Contacts' (highlighted in blue). A dropdown menu shows 'Google' (highlighted in blue) and 'Yahoo' (highlighted in blue with a cursor icon).
- Table: Shows three imported contacts:
  - David Johnson**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:34 PM', 'Connect' button.
  - John Edwards**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:34 PM', 'Connect' button.
  - David Johnson**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:34 PM', 'Connect' button.

**Screenshot 3 (Bottom): LinkedIn - Yahoo**

- Header: LinkedIn logo, Search bar, Home, My Network, Jobs, Messaging.
- Section: 'Saved contacts' (highlighted in blue), 'Yahoo ▾' (highlighted in blue).
- Content: '3 Imported Contacts' (highlighted in blue). A dropdown menu shows 'Google' (highlighted in blue) and 'Yahoo' (highlighted in blue with a cursor icon).
- Table: Shows three imported contacts:
  - David Johnson**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:34 PM', 'Connect' button.
  - John Edwards**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:34 PM', 'Connect' button.
  - David Johnson**: Profile picture, 'Saved contacts & Connections' link, 'Created: 12:34 PM', 'Connect' button.

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

183. LinkedIn has directly infringed, and continues to directly infringe, the claims of the ‘395 Patent, including at least those noted above, including by making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a). Further, including at least to the extent LinkedIn provides and/or supplies software running on a user’s computer, the direct infringement of users that occurs in connection with LinkedIn’s applications and/or web services occurs under the direction or control of LinkedIn.

184. Additionally, or in the alternative, since receiving notice of the ‘395 patent, including if necessary from this suit, LinkedIn has induced, and continues to induce, infringement of the ‘395 Patent in this judicial district, and elsewhere, by actively inducing direct infringement of the ‘395 Patent, including by knowingly and actively aiding or abetting infringement by users, by and through at least instructing and encouraging the use of the LinkedIn products and software noted herein, including the LinkedIn Application system. Such aiding and abetting comprises providing software, web servers, and/or instructions regarding the use and/or operation of the LinkedIn Application system, applications, and web servers in an infringing manner. Such induced infringement has occurred since LinkedIn became aware of the ‘395 Patent, at a minimum, as noted above, and the knowledge and awareness that such actions by users comprise infringement of the ‘395 Patent.

185. LinkedIn has had at least constructive notice of the ‘395 Patent since at least its issuance. LinkedIn will have been on actual notice of the ‘395 Patent since, at the latest, the service of this Complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the ‘395 Patent.

186. The LinkedIn Application system clearly meets the asserted claim limitations in their normal and expected usage. On information and belief, normal and expected usage of the LinkedIn Application system by customers and/or end users satisfies the claim limitations for direct infringement. Further, at minimum, the provision of products, systems and/or functionalities clearly capable of such infringing usage and/or provision of instructions/specifications for such infringing usage constitutes inducement of directly infringing usage.

187. Further, as noted above, LinkedIn is being made aware of infringement of the

‘395 Patent through use of the LinkedIn Application system at least via the infringement allegations set forth herein. Such direct and induced infringement has been and remains clear, unmistakable and inexcusable. On information and belief, LinkedIn knew or should have known of the clear, unmistakable and inexcusable direct and induced infringing conduct at least receiving notice of the ‘395 Patent. Thus, on information and belief, Defendants have, since receiving notice of the ‘395 Patent, specifically intended to induce direct infringement by customers and/or end users.

188. EBT believes and contends that, at a minimum, LinkedIn’s knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the ‘395 Patent since receiving notice of its infringement of the ‘395 Patent, is necessarily willful, wanton, malicious, in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a finding of willful infringement. Accordingly, since at least receiving notice of this suit, LinkedIn has willfully infringed the ‘395 Patent.

**COUNT II – INFRINGEMENT OF U.S. PATENT No. 9,584,453**

189. Plaintiff is the owner of the ‘453 Patent and it has all substantial rights to the ‘453 Patent, including the right and standing to sue and recover damages for past, present, and future infringement of the patent.

190. Claim 1 of the ‘453 Patent covers a system comprising “a network interface; a network login engine coupled to the network interface; a network contacts database embodied in one or more non-transitory computer-readable mediums; a server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of messaging services through a high level network using the network interface

to access contact information from the plurality of messaging services, updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list, stores the aggregated contact list in a non-transitory computer-readable medium at the server, and provides the aggregated contact list including the contact information to a display device.”

191. LinkedIn has infringed, and is now infringing, the ‘453 patent, including at least claim 1, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising the practicing, without authority from Plaintiff, systems and methods for obtaining and aggregating contact information from a plurality of messaging services providers via LinkedIn’s LinkedIn Application system, including as claimed in the ‘453 asserted claims. On information and belief, LinkedIn practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising [www.linkedin.com](http://www.linkedin.com).

192. Without limitation, the accused system comprising the LinkedIn Application system comprises a network interface; a network login engine coupled to the network interface; a network contacts database embodied in one or more non-transitory computer-readable mediums; a server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of messaging services through a high level network using the network interface to access contact information from the plurality of messaging services, updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list, stores the aggregated contact list in a non-transitory computer-readable medium at the server, and provides the aggregated contact list including the contact information to a display device.

193. For example, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods permits a user to login via the LinkedIn Application to a first and second messaging service provider, obtain a first contact list from the first messaging service provider and a second contact list from the second messaging service provider, and update the network contacts.

194. The LinkedIn Application comprises a network interface. For example, the LinkedIn Application comprises ethernet, fiber, and/or other network connectivity for facilitating LAN, WAN, and/or other network activity:

The image displays three separate screenshots of a browser's developer tools Network tab, specifically focusing on the Headers section. Each screenshot shows a request to a LinkedIn API endpoint with the following details:

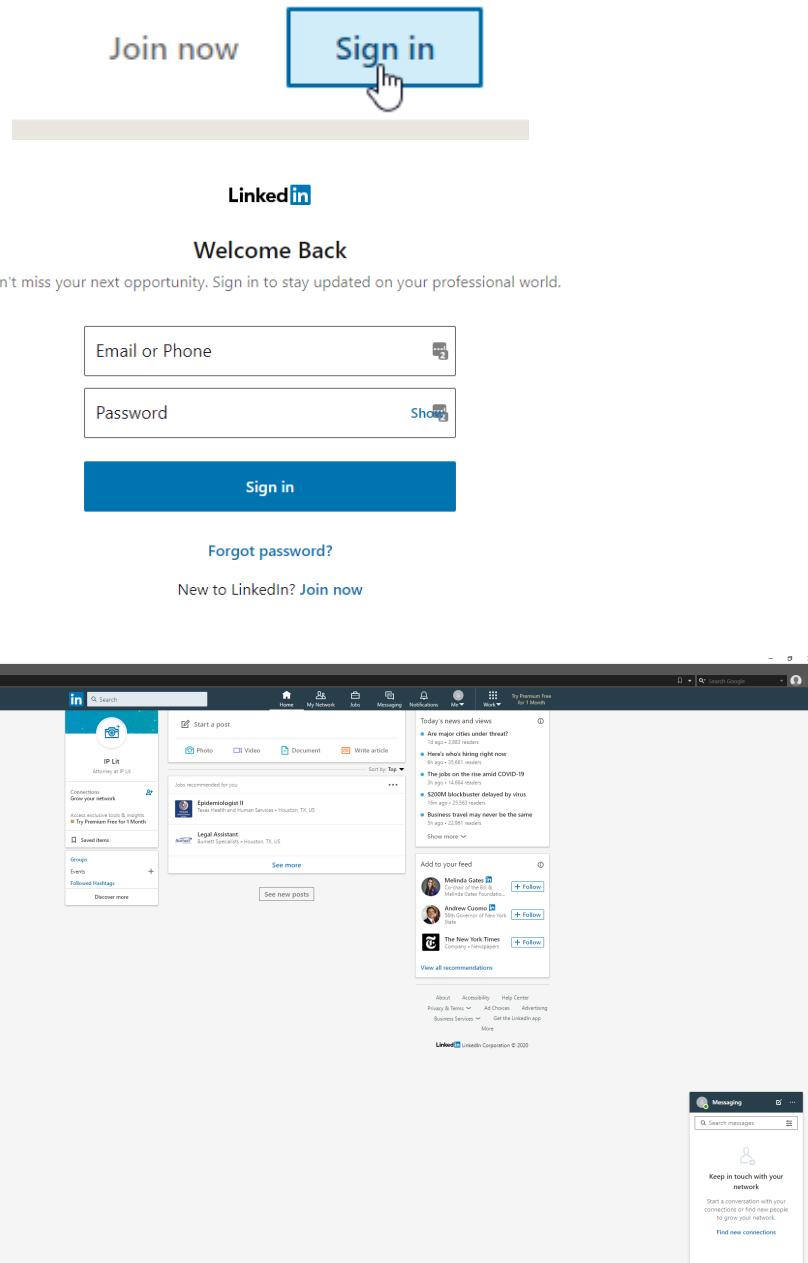
- Request URL:** `https://www.linkedin.com/voyager/api/voyagerGrowthEmailImportTask`
- Request Method:** POST
- Status Code:** 201 (indicated by a green circle)
- Remote Address:** [2620:1ec:21::14]:443
- Referrer Policy:** no-referrer-when-downgrade

The screenshots are vertically stacked, each showing a different request with identical parameters.

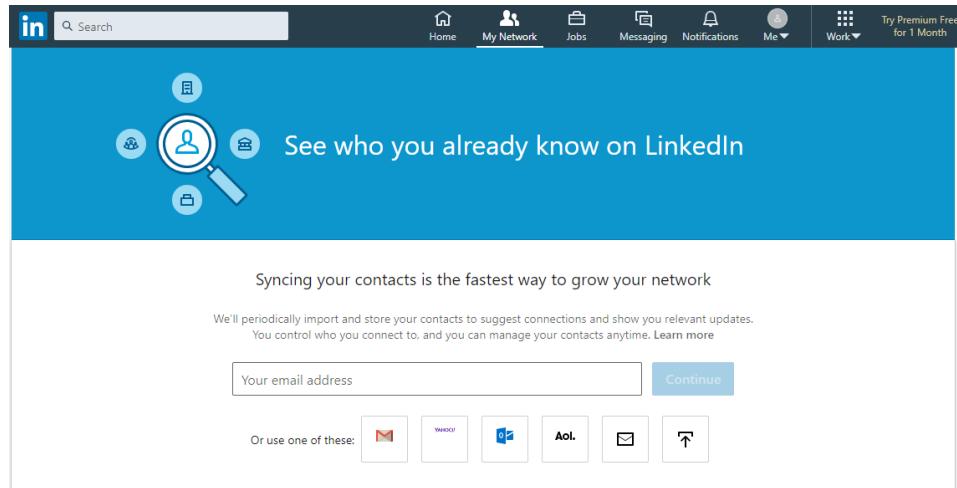
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/>

195. The LinkedIn Application comprises a network login engine coupled to the network interface. For example, the LinkedIn Application comprises the LinkedIn login webservice comprising an engine for use by the user to log into the user's LinkedIn account

and/or the user's social media or other networks coupled to the LinkedIn servers:



See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>



Or use one of these:



See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>

#### Manage synced sources

To learn more about our privacy and security practices, visit our [Privacy Policy](#) page.  
[Learn more in the Help Center](#)

Contacts	Remove all
Syncing your contact information helps you keep in touch with your most important connections, so you always know the right times to reach out.	
Google	Sync
Outlook - Personal	<a href="#">Learn how</a>
Outlook - Work	<a href="#">Learn how</a>
Phone contacts	<a href="#">Learn how</a>

Calendar	Remove all
Syncing your calendar lets you see your meeting history with people, and we'll tell you who's in your next meeting.	
Google	Sync
Phone calendar	<a href="#">Learn how</a>

[← Back to Manage Your Contacts](#)

Contacts Remove all

Syncing your contact information helps you keep in touch with your most important connections, so you always know the right times to reach out.

 Sync

See, e.g., LinkedIn User Manage Syncing page  
<https://www.linkedin.com/mynetwork/settings/manage-syncing/>

Headers Preview Response Initiator Timing Cookies

General

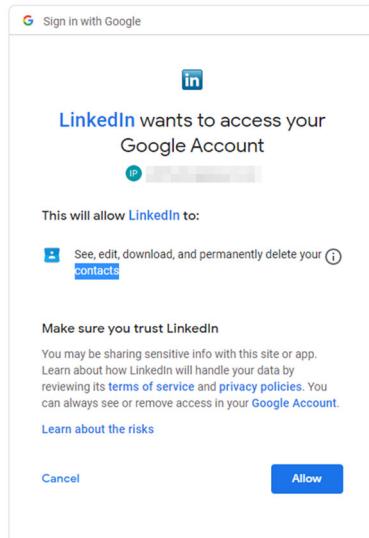
Request URL: https://accounts.google.com/o/oauth2/v2/auth?client\_id=802926523257.apps.googleusercontent.com&redirect\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds+profile+email&access\_type=offline&response\_type=code&state=f985e48a-be6f-4dd1-b728-102900db9c4b&prompt=consent&flowName=GeneralOAuthFlow

Request Method: GET

Status Code: 200

Remote Address: [2607:f8b0:4000:815::200d]:443

Referrer Policy: no-referrer-when-downgrade



See, e.g., Google Sign-In Request page at  
[https://accounts.google.com/o/oauth2/v2/auth/identifier?client\\_id=802926523257.apps.googleusercontent.com&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email](https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email)

LinkedIn

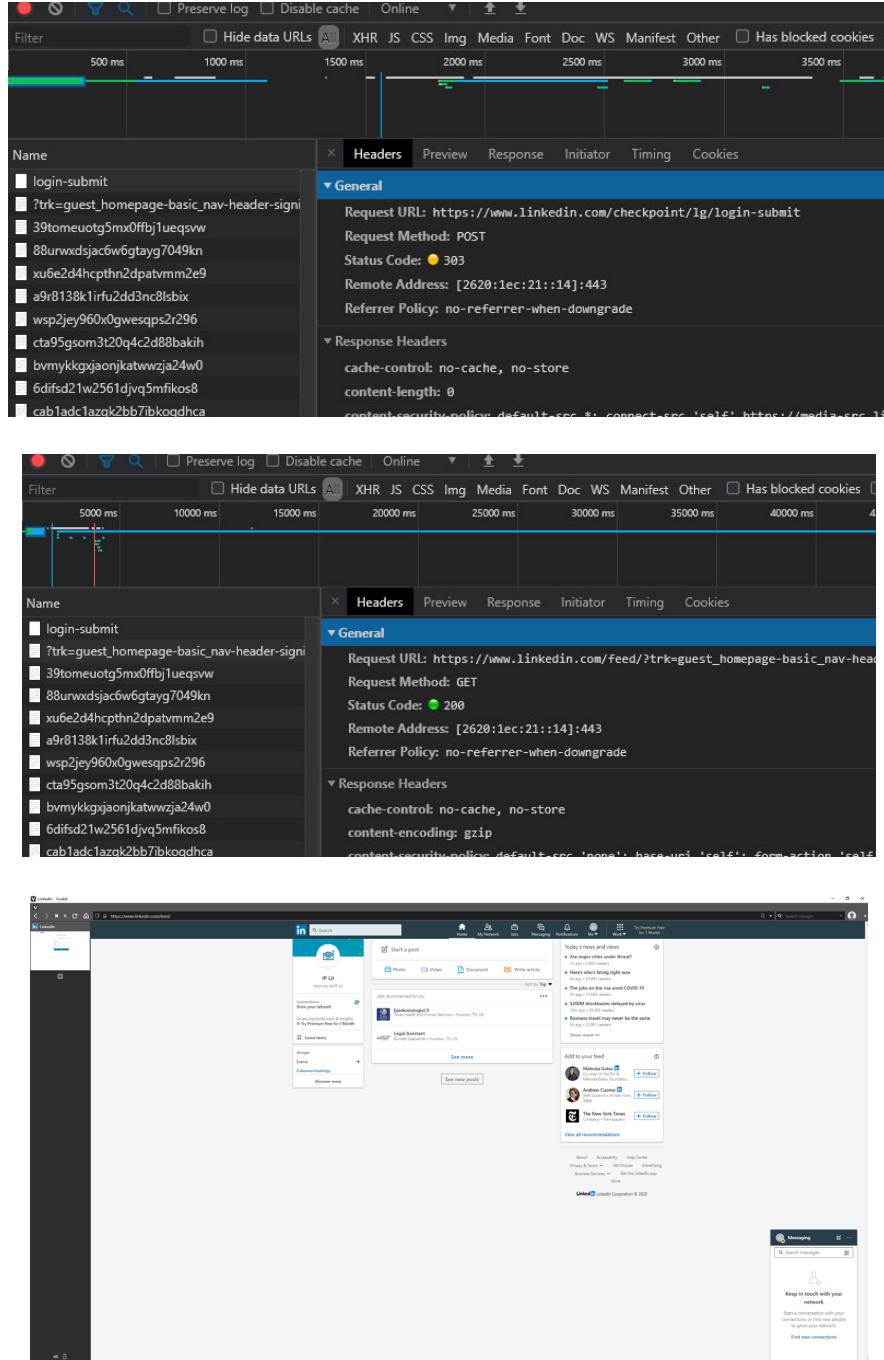
Welcome Back

Don't miss your next opportunity. Sign in to stay updated on your professional world.

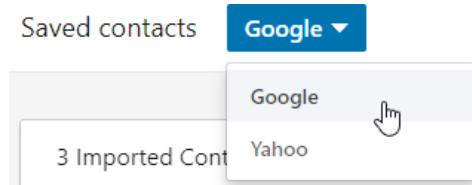
Email or Phone	
Password	Show 
<input type="button" value="Sign in"/>	

[Forgot password?](#)

New to LinkedIn? [Join now](#)



See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>



The screenshot shows the LinkedIn desktop interface. At the top, there's a navigation bar with the LinkedIn logo, a search bar, and links for Home, My Network, Jobs, and Messaging. Below the navigation bar, a blue header bar says "Saved contacts" and "Google". Underneath, a section titled "3 Imported Contacts" is displayed, sorted by "Recently added". Three contacts are listed:

- John Edwards (Created: 12:24 PM) with a "Connect" button.
- John Edwards (Created: 12:24 PM) with a "Connect" button.
- John Edwards (Created: 12:24 PM) with a "Pending" status.

A dropdown menu is open over the "Google" link in the header, showing options: "Google" (disabled) and "Yahoo" (selected).

This screenshot shows the same LinkedIn interface after switching the provider to "Yahoo". The "Saved contacts" header now says "Yahoo". The "3 Imported Contacts" section remains the same, showing three contacts from Google. A dropdown menu is open over the "Yahoo" link in the header, showing options: "Google" (disabled) and "Yahoo" (selected).

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

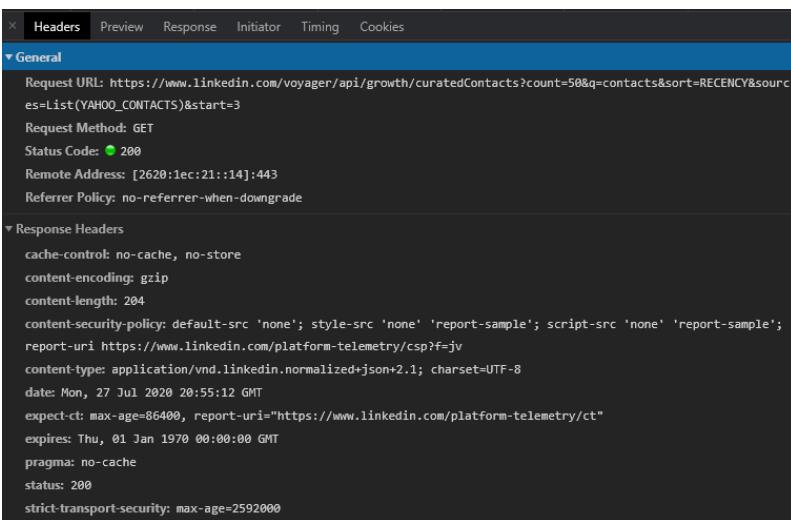
This screenshot shows a sidebar menu titled "Manage my network". It includes the following options:

- Connections
- Contacts (highlighted with a cursor icon, showing 3 items)
- People I Follow
- Groups
- Events
- Pages
- Newsletters
- Hashtags

See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>



The screenshot shows the LinkedIn 'Saved contacts' section. At the top, there's a dropdown menu with 'Yahoo' selected. Below it, a modal window displays '3 Imported Cont' (Contacts) with a 'Yahoo' button highlighted by a cursor. The main interface shows a list of three imported contacts, each with a profile picture, name, company, creation date, and a 'Connect' button.



The screenshot shows the Network tab of a browser developer tools window titled 'Headers'. It details a network request to the URL `https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List(YAHOO_CONTACTS)&start=3`. The request method is GET, status code is 200, and the response includes various HTTP headers like Cache-Control, Content-Encoding, Content-Length, Content-Security-Policy, Date, Expect-CT, Expires, Pragma, Status, and Strict-Transport-Security.

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

196. The LinkedIn Application comprises a network contacts database embodied in one or more non-transitory computer readable mediums. For example, the LinkedIn Application comprises an address book comprising at least the user's LinkedIn contacts embodied in, for example, a hard drive and/or other ROM and/or, at a minimum, RAM, for storing the user's contact list in said database:

## LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

**Note:** Your [contacts](#) and [connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

## Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically [import and store](#) details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

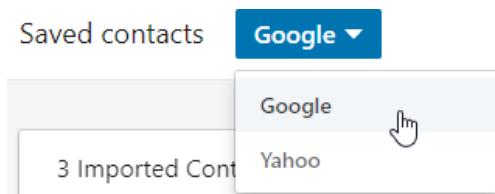
See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

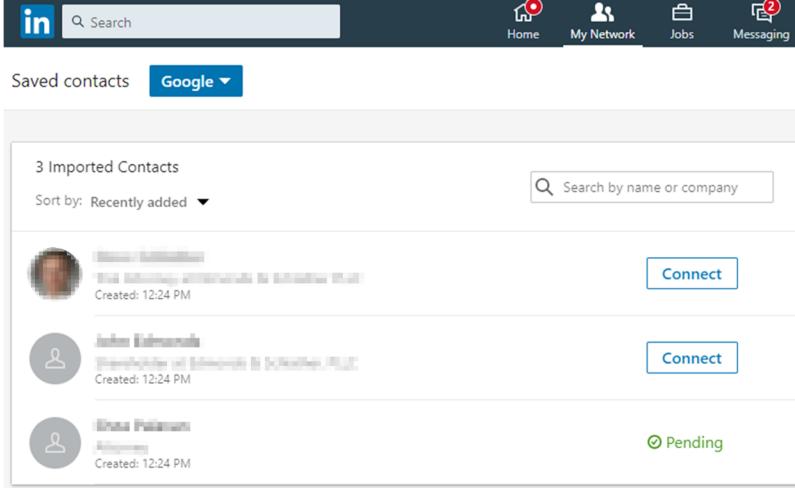
### Contact and Calendar Information

We receive personal data (including contact information) about you when others [import or sync their contacts or calendar with our Services](#), associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>





Saved contacts Google ▾

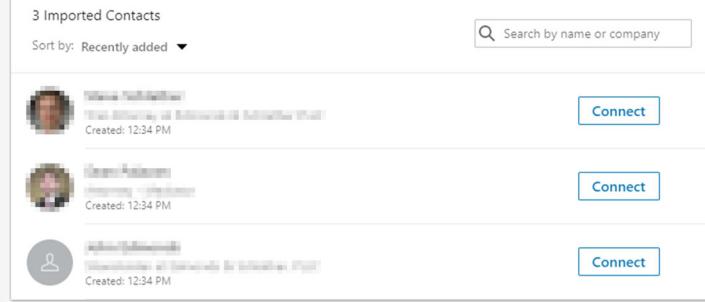
3 Imported Contacts  
Sort by: Recently added ▾

Profile Picture	Name	Description	Action
	John Edwards	Imported via Google (1 contact)	<span style="border: 1px solid #0072bc; padding: 2px 5px;">Connect</span>
	John Edwards	Imported via Google (1 contact)	<span style="border: 1px solid #0072bc; padding: 2px 5px;">Connect</span>
	John Edwards	Imported via Google (1 contact)	<span style="color: green; border: 1px solid #0072bc; padding: 2px 5px;">Pending</span>


Saved contacts Yahoo ▾

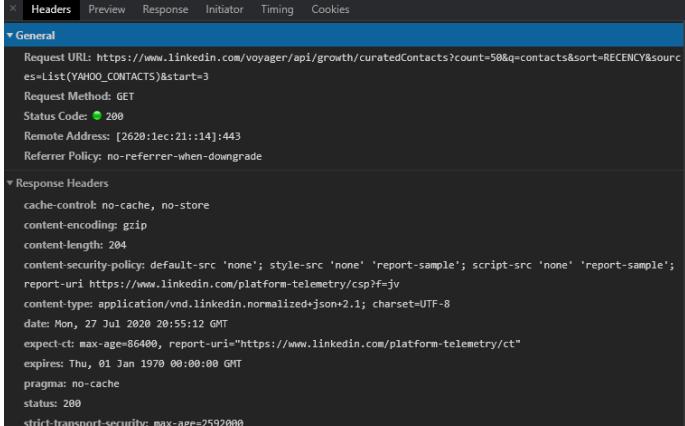
3 Imported Cont  
Google  
Yahoo

Saved contacts Yahoo ▾

3 Imported Contacts  
Sort by: Recently added ▾

Profile Picture	Name	Description	Action
	John Edwards	Imported via Yahoo (1 contact)	<span style="border: 1px solid #0072bc; padding: 2px 5px;">Connect</span>
	John Edwards	Imported via Yahoo (1 contact)	<span style="border: 1px solid #0072bc; padding: 2px 5px;">Connect</span>
	John Edwards	Imported via Yahoo (1 contact)	<span style="border: 1px solid #0072bc; padding: 2px 5px;">Connect</span>

Headers Preview Response Initiator Timing Cookies

General

Request URL: https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List(YAHOO\_CONTACTS)&start=3  
 Request Method: GET  
 Status Code: 200  
 Remote Address: [2620:1ec:21:14]:443  
 Referer Policy: no-referrer-when-downgrade

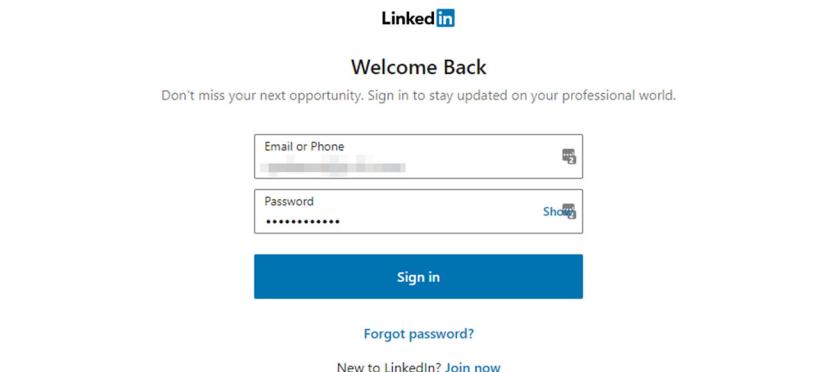
Response Headers

```
cache-control: no-cache, no-store
content-encoding: gzip
content-length: 204
content-security-policy: default-src 'none'; style-src 'none' 'report-sample'; script-src 'none' 'report-sample';
report-uri: https://www.linkedin.com/platform-telemetry/csp?f=jv
content-type: application/vnd.linkedin.normalized+json+2.1; charset=UTF-8
date: Mon, 27 Jul 2020 20:55:12 GMT
expect-ct: max-age=86400, report-uri="https://www.linkedin.com/platform-telemetry/ct"
expires: Thu, 01 Jan 1970 00:00:00 GMT
pragma: no-cache
status: 200
strict-transport-security: max-age=2592000
```

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

197. The LinkedIn Application comprises a server coupled to the network contacts

database. For example, the LinkedIn Application comprises a server connected to the LinkedIn contact database:



The screenshot shows the LinkedIn login page with fields for 'Email or Phone' and 'Password'. Below the password field is a 'Sign in' button and a 'Forgot password?' link. At the bottom, there's a 'New to LinkedIn? Join now' link.

**Network Traffic Analysis:**

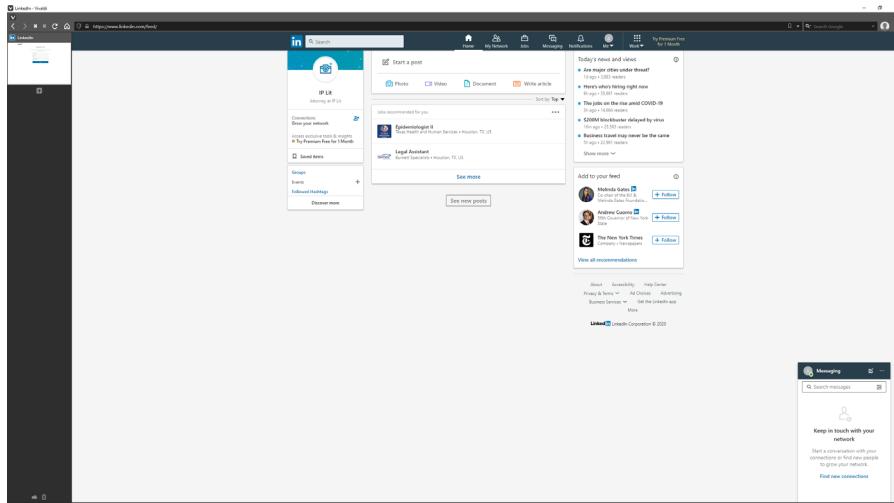
- Request 1 (POST):** Request URL: <https://www.linkedin.com/checkpoint/lg/login-submit>. Request Method: POST. Status Code: 303. Remote Address: [2620:1ec:21::1]:443. Referrer Policy: no-referrer-when-downgrade.
- Request 2 (GET):** Request URL: [https://www.linkedin.com/feed/?trk=guest\\_homepage-basic\\_nav-header-signin](https://www.linkedin.com/feed/?trk=guest_homepage-basic_nav-header-signin). Request Method: GET. Status Code: 200. Remote Address: [2620:1ec:21::1]:443. Referrer Policy: no-referrer-when-downgrade.

**Headers for Request 1:**

- login-submit
- ?trk=guest\_homepage-basic\_nav-header-signin
- 39tomeuotg5mx0ffbjlueqsvw
- 88urwdxdsjac6w6gtayg7049kn
- xuf6e2d4hcpthn2dpavmm2e9
- a9r8138k1rfu2dd3nc8lsbix
- wsp2jey960x0gvesqps2r296
- cta95gsm3t20q4c2d88bakih
- bvmykkpxjaonjkatwwza24w0
- 6dfsf21wz561djvq5nfikos8
- cab1adc1azqk2bb7ibkoqdha

**Headers for Request 2:**

- login-submit
- ?trk=guest\_homepage-basic\_nav-header-signin
- 39tomeuotg5mx0ffbjlueqsvw
- 88urwdxdsjac6w6gtayg7049kn
- xuf6e2d4hcpthn2dpavmm2e9
- a9r8138k1rfu2dd3nc8lsbix
- wsp2jey960x0gvesqps2r296
- cta95gsm3t20q4c2d88bakih
- bvmykkpxjaonjkatwwza24w0
- 6dfsf21wz561djvq5nfikos8
- cab1adc1azqk2bb7ibkoqdha



See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>

198. The LinkedIn Application comprises a contact aggregation engine coupled to the network login engine and the network contacts database. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager which provides LinkedIn's servers to aggregate the user's contacts coupled to the LinkedIn login engine and to the storage medium comprising the user's LinkedIn contacts:

#### LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

**Note:** Your [contacts and connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

#### Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically [import and store](#) details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

#### Contact and Calendar Information

We receive personal data (including contact information) about you when others [import or sync their contacts or calendar with our Services](#), associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

#### Syncing Contacts from Other Address Books and Sources

[LinkedIn Contacts Manager](#) can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

**Note:** If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.

To sync contacts:

1. Click the  My Network icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  Manage synced contacts near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

##### Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the  My Network icon at the top of your LinkedIn homepage.
  2. Click **Contacts** under **Manage my network** on the left rail.
  3. Click  Add more contacts on the right rail.
- Note:** You'll be redirected to a page where you can enter the source you want to import the contacts from.

**Note:** If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

**Important:** Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about [deleting imported contacts](#) and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>



**Welcome Back**

Don't miss your next opportunity. Sign in to stay updated on your professional world.



 [Show](#)

**Sign in**

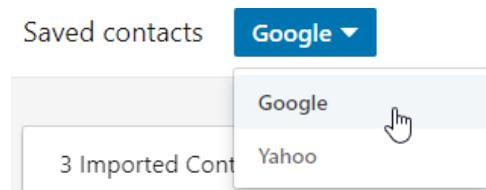
[Forgot password?](#)

New to LinkedIn? [Join now](#)

The figure consists of three vertically stacked screenshots of a browser's developer tools Network tab, illustrating network traffic to LinkedIn's servers.

- Top Screenshot:** Shows a POST request to `https://www.linkedin.com/checkpoint/lg/login-submit`. The Headers panel shows the following details:
  - Request URL: `https://www.linkedin.com/checkpoint/lg/login-submit`
  - Request Method: POST
  - Status Code: 303
  - Remote Address: [2620:1ec:21::14]:443
  - Referrer Policy: no-referrer-when-downgrade
 The Cookies panel lists several session cookies, including `login-submit`, `?trk=guest_homepage-basic_nav-header-signin`, and various long alphanumeric strings.
- Middle Screenshot:** Shows a GET request to `https://www.linkedin.com/feed/?trk=guest_homepage-basic_nav-header-signin`. The Headers panel shows:
  - Request URL: `https://www.linkedin.com/feed/?trk=guest_homepage-basic_nav-header-signin`
  - Request Method: GET
  - Status Code: 200
  - Remote Address: [2620:1ec:21::14]:443
  - Referrer Policy: no-referrer-when-downgrade
 The Cookies panel lists the same set of session cookies as the top screenshot.
- Bottom Screenshot:** A screenshot of the LinkedIn desktop website at `https://www.linkedin.com/feed/`. The page displays a news feed with posts from users like "Epidemiologist II" and "Legal Assistant". It includes sections for "Start a post", "Jobs recommended for you", and "Add to your feed". The right sidebar features a messaging interface.

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>



This screenshot shows the LinkedIn desktop website interface. At the top, there's a navigation bar with the LinkedIn logo, a search bar, and links for Home, My Network, Jobs, and Messaging. Below the navigation bar, a dropdown menu is open, showing 'Saved contacts' and 'Google'. The main content area displays a list titled '3 Imported Contacts' sorted by 'Recently added'. Each contact entry includes a profile picture, name, company, creation date, and a 'Connect' button.

Profile Picture	Name	Company	Created	Action
[Placeholder]	John Doe	ABC Company	Created: 12:24 PM	Connect
[Placeholder]	Jane Smith	XYZ Corporation	Created: 12:24 PM	Connect
[Placeholder]	David Johnson	DEF Industries	Created: 12:24 PM	Pending

This screenshot shows a context menu on the LinkedIn desktop website. A dropdown menu is open from a 'Yahoo' button, listing 'Google' and 'Yahoo'. The 'Yahoo' option is highlighted with a mouse cursor. The background shows the 'Saved contacts' section with '3 Imported Contacts' listed.

This screenshot shows the LinkedIn desktop website interface again, but this time it's under a different account. The dropdown menu shows 'Saved contacts' and 'Yahoo'. The main content area displays the same '3 Imported Contacts' list as the previous screenshots.

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

This screenshot shows the LinkedIn user's 'MyNetwork' page. It features a sidebar with options: 'Manage my network', 'Connections', 'Contacts' (which is selected and has a count of 3), 'People I Follow', 'Groups', 'Events', 'Pages', 'Newsletters', and 'Hashtags'. The main content area is currently empty.

See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>



Saved contacts    Yahoo ▾

Google

3 Imported Cont

Yahoo

Saved contacts    Yahoo ▾

3 Imported Contacts

Sort by: Recently added ▾

Search by name or company

		Connect
	Recent Addition to LinkedIn Network	Created: 12:34 PM
	Recent Addition to LinkedIn Network	Created: 12:34 PM
	Recent Addition to LinkedIn Network	Created: 12:34 PM

Headers   Preview   Response   Initiator   Timing   Cookies

General

Request URL: [https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List\(YAHOO\\_CONTACTS\)&start=3](https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List(YAHOO_CONTACTS)&start=3)  
 Request Method: GET  
 Status Code: 200  
 Remote Address: [2620:1ec:21::14]:443  
 Referrer Policy: no-referrer-when-downgrade

Response Headers

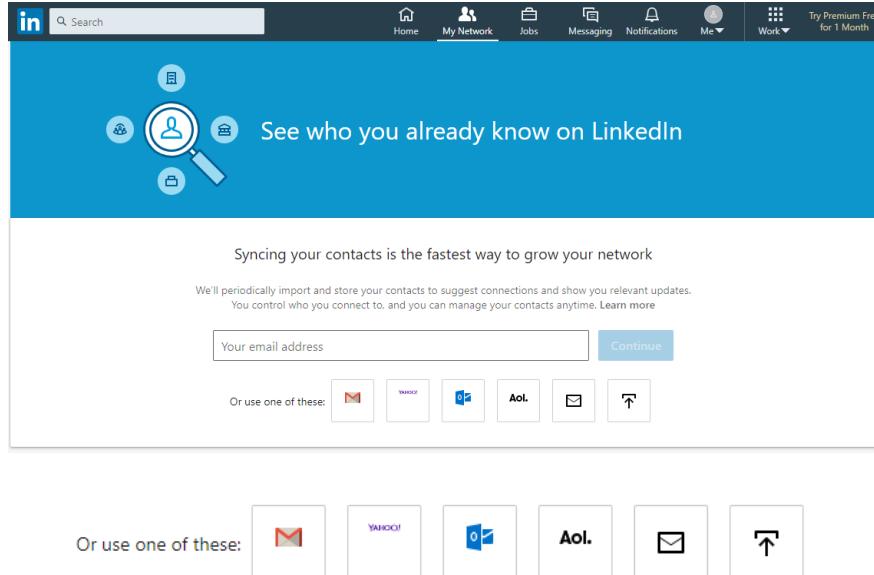
```

cache-control: no-cache, no-store
content-encoding: gzip
content-length: 204
content-security-policy: default-src 'none'; style-src 'none' 'report-sample'; script-src 'none' 'report-sample';
report-uri https://www.linkedin.com/platform-telemetry/csp?f=jv
content-type: application/vnd.linkedin.normalized+json+2.1; charset=UTF-8
date: Mon, 27 Jul 2020 20:55:12 GMT
expect-ct: max-age=86400, report-uri="https://www.linkedin.com/platform-telemetry/ct"
expires: Thu, 01 Jan 1970 00:00:00 GMT
pragma: no-cache
status: 200
strict-transport-security: max-age=2592000
  
```

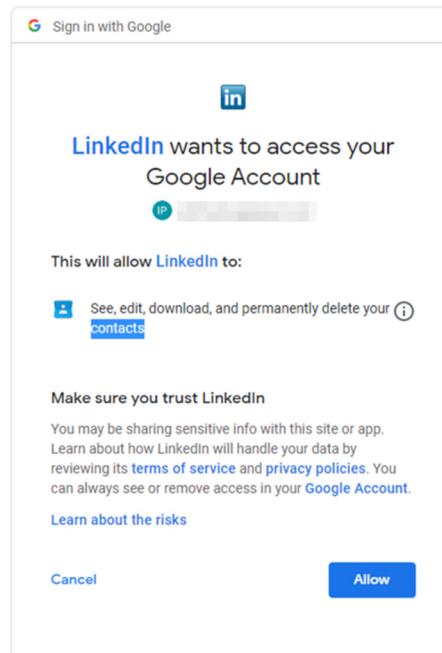
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

199. In operation, the contact aggregation engine (*see* above) controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of messaging service services through a high level network. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including facilitating the user logging into and

associating the user's various social media and/or other networks, including via APIs of LinkedIn and/or APIs of the user's social media and/or other networks:



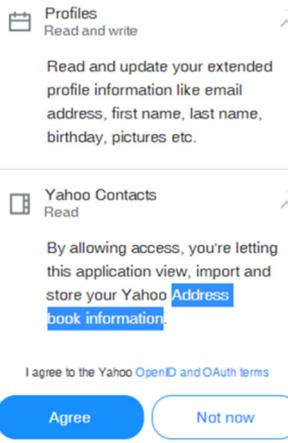
See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>



See, e.g., Google Sign-In Request page at  
[https://accounts.google.com/o/oauth2/v2/auth/identifier?client\\_id=802926523257.apps.googleusercontent.com&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email](https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email)

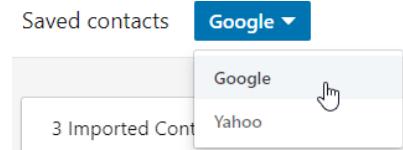
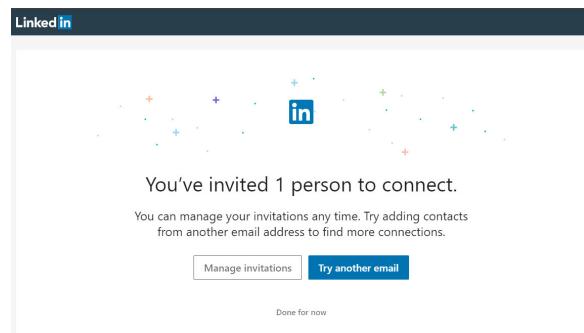
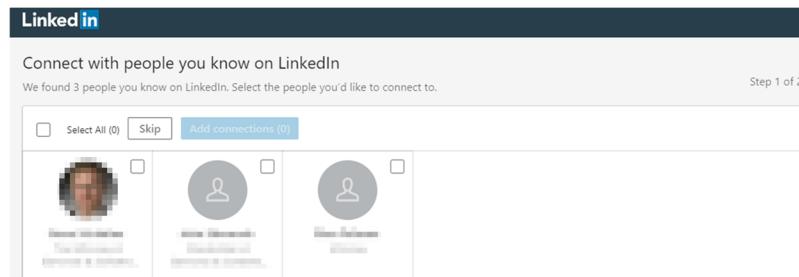
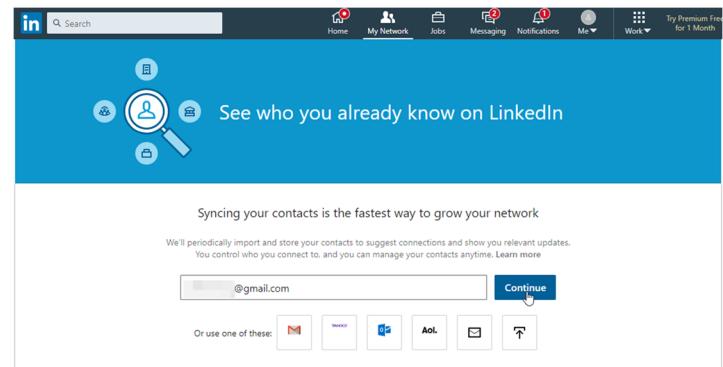
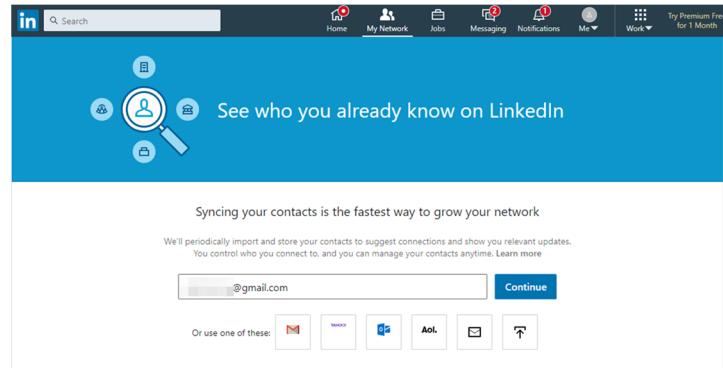
Hi, [REDACTED]

By agreeing, you'll sign in to  
Linkedin with your Yahoo account  
and allow Linkedin to access:



See, e.g., Yahoo! Sign-In Request page at  
[https://api.login.yahoo.com/oauth2/request\\_auth\\_fe?client\\_id=dj0yJmk9S0c3Y2RXVmwlRTQxJmQ9WVdrOVIYWmxXbWxDTXpBbWNHbzINQS0tJnM9Y29uc3VtZXJzZWNyZXQmeD02MA--&redirect\\_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=openid%20sdpp-w%20sdct-r&sec=true&response\\_type=code&state=5ef8e091-7714-423c-a57bf607710145fb&nonce=6612439201265914688&guccounter=1&guce\\_referrer=aHR0cHM6Ly9sb2dpbi55YWhvby5jb20v&guce\\_referrer\\_sig=AQAAADx0ox5D1DZWYJqN1T31M8wjeY72WHnm5USj7t1-994XN4dtTs9wXZPxbFnm9LuyM26LTbfNgpUgeh89NDgcnS2xsBPJoIiDu6onlExDhf7tsoQEJKVkkqOu2ac4W3pj3P34o\\_G23xo8DLX4pMAqIKiP6olGZjKhodFpcMc2yR](https://api.login.yahoo.com/oauth2/request_auth_fe?client_id=dj0yJmk9S0c3Y2RXVmwlRTQxJmQ9WVdrOVIYWmxXbWxDTXpBbWNHbzINQS0tJnM9Y29uc3VtZXJzZWNyZXQmeD02MA--&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=openid%20sdpp-w%20sdct-r&sec=true&response_type=code&state=5ef8e091-7714-423c-a57bf607710145fb&nonce=6612439201265914688&guccounter=1&guce_referrer=aHR0cHM6Ly9sb2dpbi55YWhvby5jb20v&guce_referrer_sig=AQAAADx0ox5D1DZWYJqN1T31M8wjeY72WHnm5USj7t1-994XN4dtTs9wXZPxbFnm9LuyM26LTbfNgpUgeh89NDgcnS2xsBPJoIiDu6onlExDhf7tsoQEJKVkkqOu2ac4W3pj3P34o_G23xo8DLX4pMAqIKiP6olGZjKhodFpcMc2yR)

200. In operation, the contact aggregation engine (*see* above) uses the network interface to access contact information from the plurality of messaging services. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager obtaining and importing the contacts of the user's various added networks onto LinkedIn's servers via connection to the servers of these other networks, including via APIs of LinkedIn and/or these other networks, after the user logs in to the networks:



The screenshot shows the LinkedIn desktop interface. At the top, there's a navigation bar with icons for Home, My Network, Jobs, and Messaging. Below that, a search bar and a 'Saved contacts' dropdown are visible. The main content area is titled '3 Imported Contacts' and shows three entries. Each entry includes a small profile picture, the contact's name, their current position at a company, and a 'Connect' button. The third contact entry also has a green 'Pending' status indicator next to its 'Connect' button.

See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

201. In operation, the contact aggregation engine (*see* above) updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including creating and maintaining a combined, aggregated contact list using the user's imported contacts via the LinkedIn Contacts Manager, including maintaining a combined, aggregated contact list using the user's imported contacts via the LinkedIn Contacts Manager:

#### LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

**Note:** Your [contacts and connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

## Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically [import and store](#) details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

### Contact and Calendar Information

We receive personal data (including contact information) about you when others [import or sync their contacts or calendar with our Services](#), associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

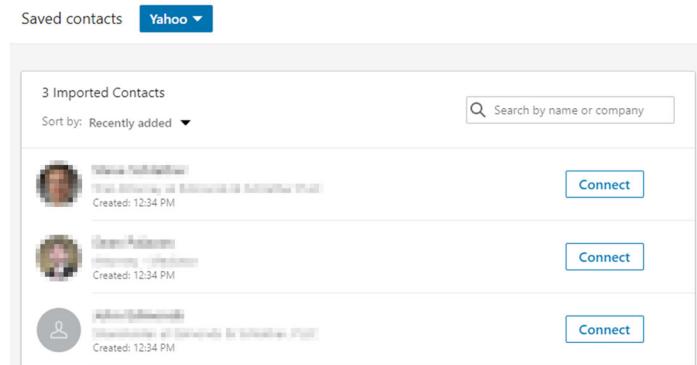
Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

The screenshot shows the LinkedIn desktop interface. At the top, there's a navigation bar with the LinkedIn logo, a search bar, and links for Home, My Network, Jobs, and Messaging. Below the navigation is a dropdown menu for 'Saved contacts' which is currently set to 'Google'. A sub-menu is open, showing 'Google' and 'Yahoo' with a cursor hovering over 'Yahoo'. The main content area displays a list of '3 Imported Contacts' sorted by 'Recently added'. Each contact entry includes a profile picture, name, company, location, and creation date (all listed as 'Created: 12:24 PM'). To the right of each entry is a 'Connect' button. The bottom of the list shows a contact with a status of 'Pending'.

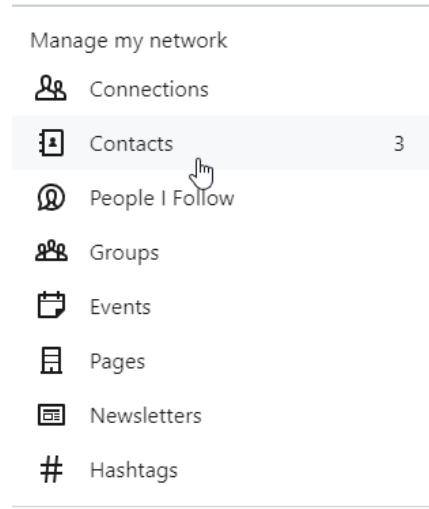
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

This screenshot shows the same LinkedIn desktop interface as the previous one, but with a different selection in the 'Saved contacts' dropdown. The dropdown is now set to 'Yahoo', and its sub-menu is open, showing 'Google' and 'Yahoo' with a cursor hovering over 'Yahoo'. The main content area below shows the same list of '3 Imported Contacts' as the previous screenshot, but the contact details are not fully visible due to the crop.

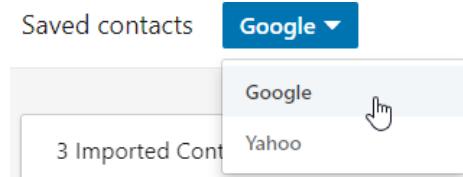


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

202. In operation, the contact aggregation engine (*see* above) stores the aggregated contact list in a non-transitory computer-readable medium at the server. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including storing the user's aggregated contact list on LinkedIn's servers:



See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>



The screenshots illustrate the LinkedIn contact aggregation process. The top two images show the 'Saved contacts' page for 'Google' and 'Yahoo' respectively, displaying three imported contacts from LinkedIn. The bottom image shows the 'Saved contacts' page for 'Yahoo' with a callout pointing to the 'Yahoo' button, indicating where the aggregation occurred.

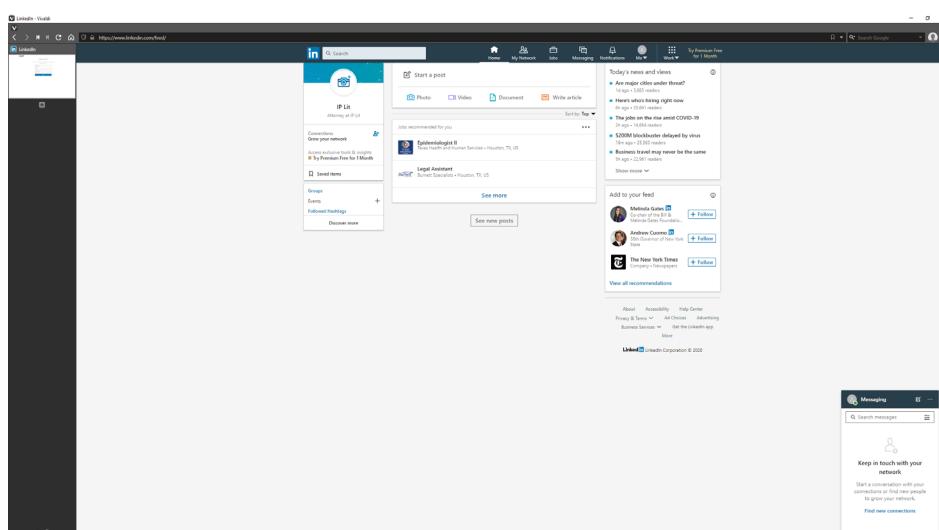
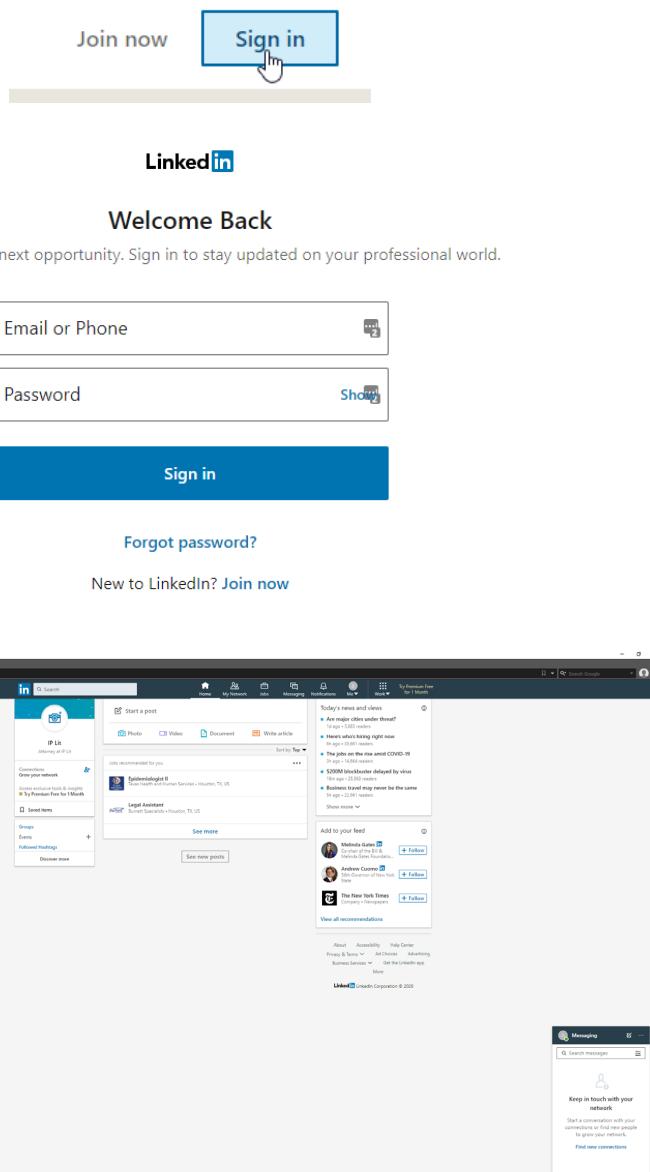
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

The screenshot shows a network request made by LinkedIn's contact aggregation engine. The request URL is `https://www.linkedin.com/voyager/api/growth/curatedContacts?count=50&q=contacts&sort=RECENTY&sources=List(YAHOO_CONTACTS)&start=3`. The request method is GET, and the status code is 200 OK. The response headers include cache-control, content-encoding, content-length, content-security-policy, report-uri, content-type, date, expect-ct, expires, pragma, status, and strict-transport-security.

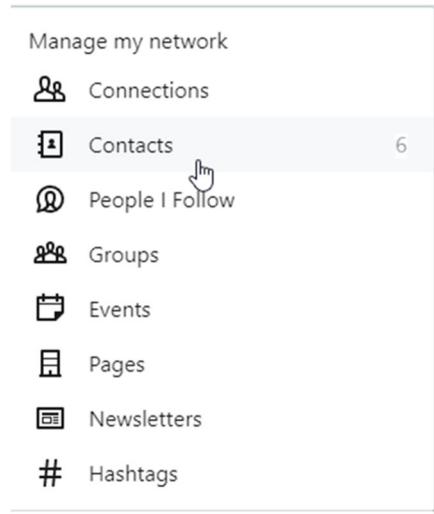
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

203. In operation, the contact aggregation engine (*see* above) provides the aggregated

contact list including the contact information to a display device. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including providing the user's aggregated contact list stored on LinkedIn's servers, which comprises the user's LinkedIn contacts and imported contacts:



See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>



See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>

Saved contacts    Google ▾

3 Imported Cont

Google    Yahoo

LinkedIn navigation bar: Home, My Network, Jobs, Messaging

Saved contacts    Google ▾

3 Imported Contacts

Sort by: Recently added ▾

Search by name or company

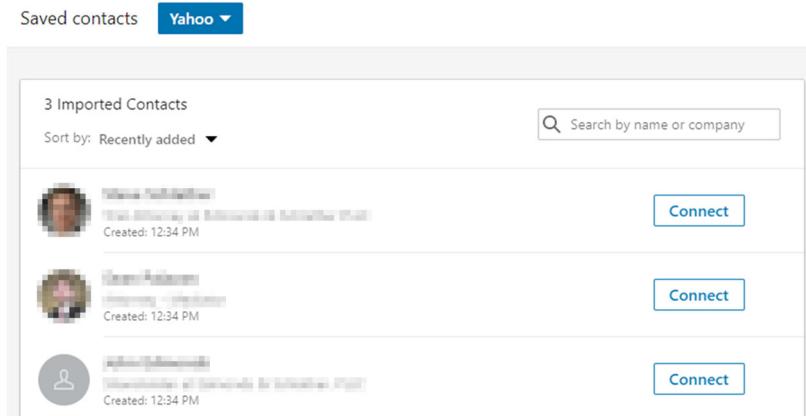
	John Edwards	Created: 12:24 PM	Connect
	John Edwards	Created: 12:24 PM	Connect
	David Palmer	Created: 12:24 PM	.Pending

Saved contacts    Yahoo ▾

3 Imported Cont

Google

Yahoo



*See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>.*

204. LinkedIn has directly infringed, and continues to directly infringe, the claims of the ‘453 Patent, including at least those noted above, including by at least making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a).

205. Additionally, or in the alternative, LinkedIn has induced, and continues to induce, infringement of the ‘453 Patent in this judicial district, and elsewhere, by actively inducing direct infringement of the ‘453 Patent, including by knowingly and actively aiding or abetting infringement by users, by and through at least instructing and encouraging the use of the LinkedIn products and software noted above, including the LinkedIn Application system. Such aiding and abetting comprises providing software, web servers, and/or instructions regarding the use and/or operation of the LinkedIn Application system, applications, and web servers in an infringing manner. Further, the direct infringement of users that occurs in connection with LinkedIn’s applications and/or web services occurs under the direction or control of LinkedIn. Such induced infringement has occurred since LinkedIn became aware of the ‘453 Patent, at a minimum, as noted above, and the knowledge and awareness that such actions by users comprise infringement of the ‘453 Patent.

206. LinkedIn has had at least constructive notice of the ‘453 Patent since at least its issuance. LinkedIn will have been on actual notice of the ‘453 Patent since, at the latest, the

service of this complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the ‘453 Patent.

207. The LinkedIn Application system clearly meets the asserted claim limitations in their normal and expected usage. On information and belief, normal and expected usage of the LinkedIn Application system by customers and/or end users satisfies the claim limitations for direct infringement. Further, at minimum, the provision of products, systems and/or functionalities clearly capable of such infringing usage and/or provision of instructions/specifications for such infringing usage constitutes inducement of directly infringing usage.

208. Further, as noted above, LinkedIn was made aware of infringement of the ‘453 patent through use of the LinkedIn Application via the infringement allegations set forth in Plaintiff’s Original Complaint, of which Defendants were aware at least as of the service of said Original Complaint. Such direct and induced infringement has been and remains clear, unmistakable and inexcusable. On information and belief, LinkedIn knew or should have known of the clear, unmistakable and inexcusable direct and induced infringing conduct at least since receiving notice of the ‘453 Patent. Thus, on information and belief, Defendants have, since receiving notice of the ‘453 Patent, specifically intended to induce direct infringement by customers and/or end users.

209. EBT believes and contends that, at minimum, LinkedIn’s knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the ‘453 Patent since receiving notice of its infringement of the ‘453 Patent, is necessarily willful, wanton, malicious, in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a finding of willful infringement. Accordingly, since at least receiving notice of this

suit, LinkedIn has willfully infringed the ‘453 Patent.

**COUNT III – INFRINGEMENT OF U.S. PATENT NO. 8,230,135**

210. Plaintiff is the owner of the ‘135 Patent and it has all substantial rights to the ‘135 Patent, including the right and standing to sue and recover damages for past, present, and future infringement of the patent.

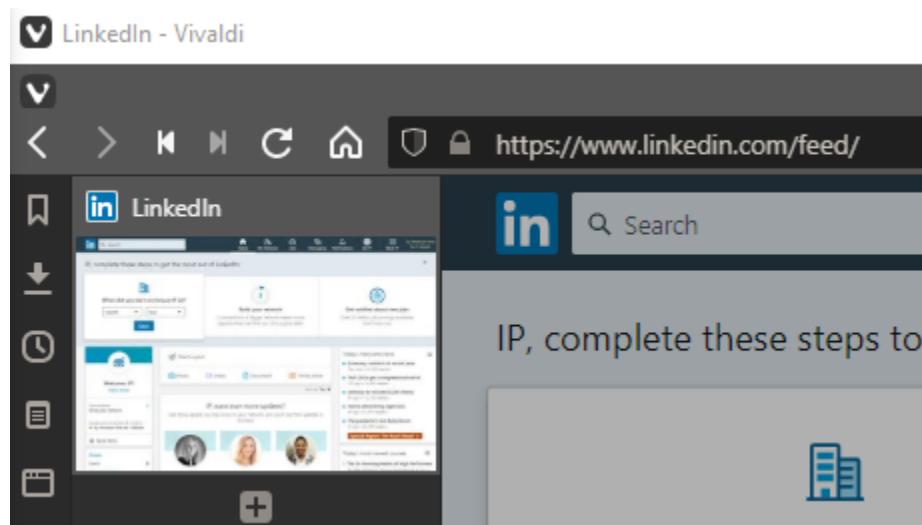
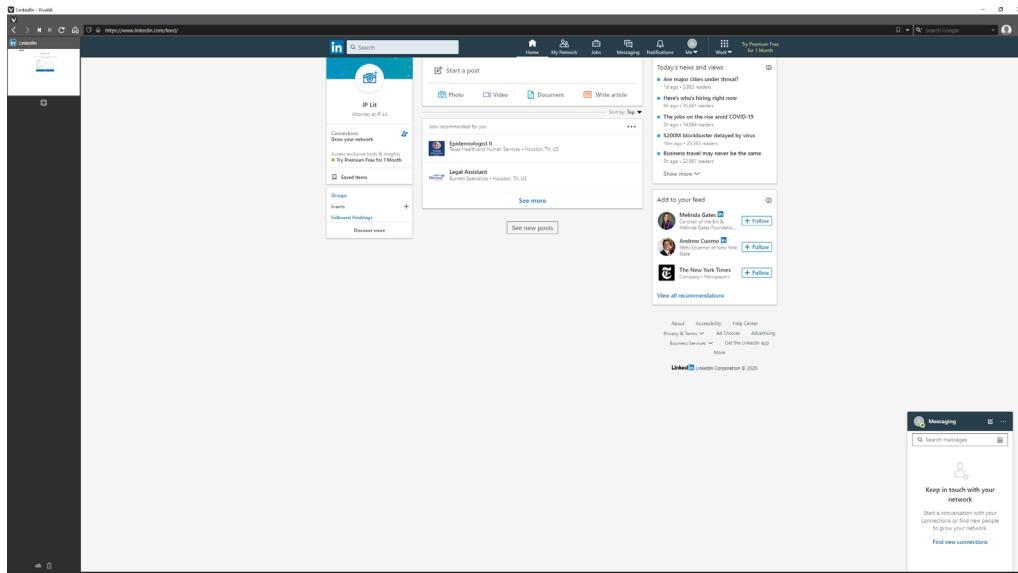
211. Claim 1 of the ‘135 Patent covers a method comprising “receiving information of an event that calls for user notification; generating an event notification for the event; associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device; providing the at least one of the plurality of character strings in the title array to a process executed by a processor; providing an alternative title based on the at least one of the plurality of character strings to the process; using the alternative title as a title in association with the process.”

212. Defendant has infringed, and is now infringing, the ‘135 patent, including at least claim 1, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising the practicing of methods and/or providing of systems, without authority from Plaintiff, for notifying a user of the occurrence of events and notification apparatuses and functionality, including notifying a user of the occurrence of an event by modification of the user’s browser title bar based on the specific event that has occurred, including as claimed in the ‘135 asserted claims. On information and belief, Defendant practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising [www.linkedin.com](http://www.linkedin.com).

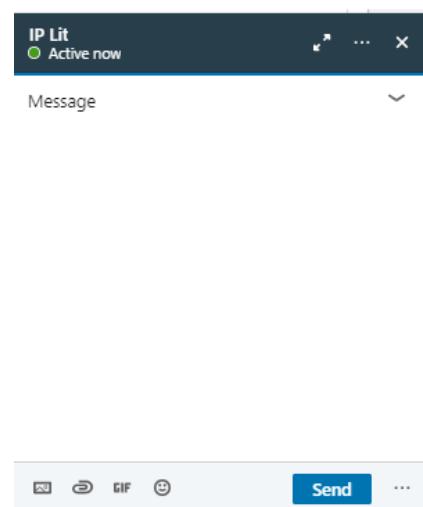
213. Without limitation, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods comprises receiving information of

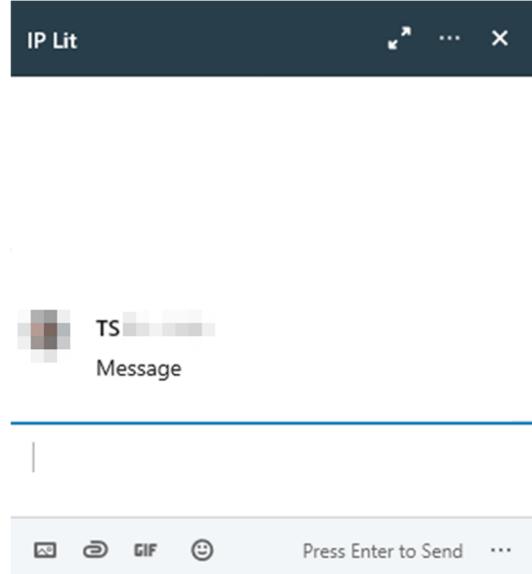
an event that calls for user notification; generating an event notification for the event; associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device; providing the at least one of the plurality of character strings in the title array to a process executed by a processor; providing an alternative title based on the at least one of the plurality of character strings to the process; using the alternative title as a title in association with the process. For example, the LinkedIn Application system permits a user's device to notify the user, via the user's browser, that the user has notifications for an event, such as a new message and/or other posting on the LinkedIn website, including via updating, modifying, and/or otherwise altering the title bar of the user's browser, including instructing the user's browser to receive notifications regarding messages the user has received from other LinkedIn users and/or posting of other LinkedIn users.

214. The LinkedIn Application receives information of an event that calls for user notification. For example, the LinkedIn Application receives information of an event that calls for user notification and it instructs and/or controls the user's browser on the user's device running code which receives code or other instructions from LinkedIn that the user has received a message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed/>





See, e.g., Second LinkedIn User Feed page at <https://www.linkedin.com/feed>

```

2113 if(o==ee){}(e,t,c),(A||B).isProxy(n)||(!e||!s.isProxy(t))&&((e,t,n, content))
2114 n=S(e,n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."), 
2115 var a=Pe(e,i) a = {value: "/voyager/api/messaging/conversations/6694727273901_4JGp_RcmK3ip_RcLnVbjpsaTpmyJyalM6
2116 ●if(null!=a)●return ●Ne(a,o,n)o = "/voyager/api/messaging/conversations/6694727273901359104/events?q=syncToken&re
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i
2118 var o,a=●(0,r.lookupDescriptor)(e,t),s=●null==a?void 0:a.set o = "/voyager/api/messaging/conversations/6694727
2119 ●if(void 0!==s&&q.●has(s)){●e[t]=n = "/voyager/api/messaging/conversations?keyVersion=LEGACY_INBOX&q=syncToken&
2120 ●return n}●if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty({●e[t]=n
2121 o!—n&&e(e,t)●else e.setUnknownProperty(t,n)
2122
2123 l=(d,e,t,c),(A||B).isProxy(n)||(!e||!s.isProxy(t))&&((e,t,n, content))
2124 n=S(e,n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."), 
2125 var a=Pe(e,i) a = {value: "urn:li:fs_event:(6694727273901359104,16694772476838248448_500)", writable: true, enumera
2126 ●if(null!=a)●return ●Ne(a,o,n)o = undefined, n = "urn:li:fs_event:(6694727273901359104,16694772476838248448_500)"
2127 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i
2128 var o,a=●(0,r.lookupDescriptor)(e,t),s=●null==a?void 0:a.set o = undefined, a = {value: "urn:li:fs_event:(669
2129 ●if(void 0!==s&&q.●has(s)){●e[t]=n = "urn:li:fs_event:(6694727273901359104,16694772476838248448_500)"
2130 ●return n}●if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty({●e[t]=n
2131 o!—n&&e(e,t)●else e.setUnknownProperty(t,n)
2132
2133 e.eBlank=He
2134 e.isPresent=function(e){return!He(e)}
2135 e.beginPropertyChanges=ce
2136 e.changeProperties=he
2137 e.endPropertyChanges=fe
2138 e.notifyPropertyChange=le
2139 e.defineProperty=Te
2140 e.createElementDescriptor=U
2141 e.nativeDescDecorator=V
2142 e.descriptorForDecorator=D
2143 e.descriptorForProperty=I
2144 e.isClassicDecorator=B
2145 e.setClassicDecorator=F
2146 e.getChainTagsForKey=we
2147 e.getProperties=function(e,t){var n={},r=arguments,i=1
2148 if(2==arguments.length&&Array.isArray(t)){i=0
2149 r=arguments[1]}for(;i<r.length;i++)n[r[i]]=S(e,r[i])
2150 return n}
2151 e.setProperties=function(e,t){if(null==t||"object"!=typeof t)r=urn:c
2152 he((function(){for(var n,r=Object.keys(t),i=0;i<r.length;i++){n=r[i] n = "_lastEventId", r = (2) ["_lastConversatio
2153 Ne(e,n,t[n])}}))
2154 return t}
2155 e.expandProperties=Oe

```

```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398   this._titles=[]
23399   this._loopNumber=0
23400   this._defaultDocumentTitle="";
23401   addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23402   this._titles=[e,this._defaultDocumentTitle]}}else this._titles.unshift(e)
23403   this._titles=this._titles.uniq()
23404   if(!this._pollerEnabled){this._pollerEnabled=!0
23405   this.pollTask("_loopTitles","","concat("document-title-poller","_").concat(Date.now()))}},resetTitle:function(){if(t
23406   this._loopNumber=0
23407   if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23408   this._pollerEnabled=11}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23409   this._titles[n-1]=this._defaultDocumentTitle
23410   var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23411   t.default&&a&&(document.title=a)
23412   this._loopNumber++
23413   this._pollerEnabled&&this.runTask(e,1500)})}
23414 define("image-edit-base/components/bg-img-edit", ["exports", "image-edit-base/components/picture-converter", "image-edit

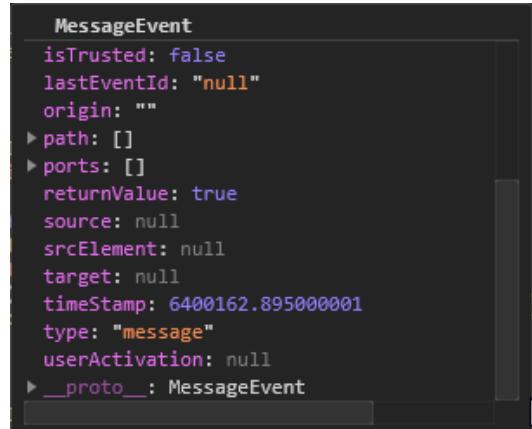
2114 n=S(e,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115   var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116   ●if(null!=a)●return ○Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117   if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+" could not be found.')})(e,t,n,i) i
2118   var o,a=○(o,r.lookupDescriptor("latestEvent") s=○null==a?void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119   ●if(void 0!==s&&q.○has(s)){○(o,t)=n = u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120   ●return n}●if(void 0==(o=e[i]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){○e[t]=n
2121   o!=n&&le(e,t)}else e.setUnknownProperty(t,n)

2114 n=S(e,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115   var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116   ●if(null!=a)●return ○Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117   if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+" could not be found.')})(e,t,n,i) i
2118   var o,a=○(o,r.lookupDescriptor("event") s=○null==a?void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119   ●if(void 0!==s&&q.○has(s)){○(o,t)=n = u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120   ●return n}●if(void 0==(o=e[i]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){○e[t]=n
2121   o!=n&&le(e,t)}else e.setUnknownProperty(t,n)

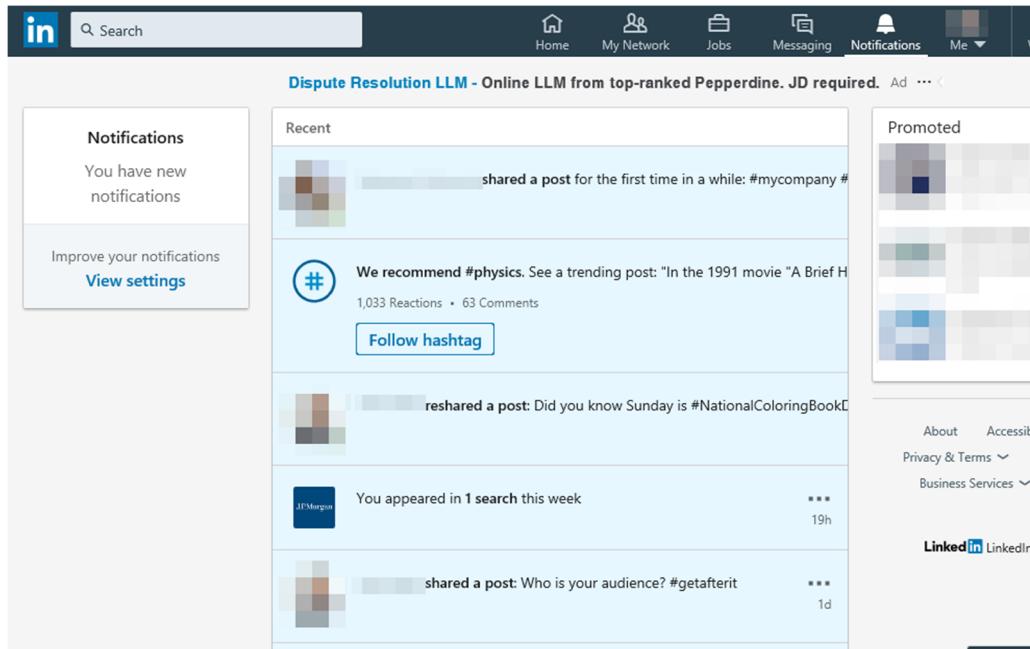
2112 void o===(n=e[t])&&(o!=0||t in e) function :=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(&&ee()){Z(i,e,t);(Array.isArray(n)||(!o.r.isEmberArray)(n))&&Z(i,e,n,"[]");(!o.r.isProxy)(n)&&Z(i,e,n,"content"))
2114 n=S(e,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115   var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116   ●if(null!=a)●return Ne(a,o,n) o = "LinkedIn", n = "(1) LinkedIn"
2117   if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+" could not be found.')})(e,t,n,i) i
2118   var o,a=○(o,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119   if(void 0==s&&q.has(s)){e[t]=n = "(1) LinkedIn"
2120   return n}if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121   o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122   return n}}function xe(){var Me=(function(e){(o,t.inheritsLoose)(i,e)
2123   function i(t){var n;(n=e.call(this)||this).__volatile__=1
2124   n. readOnly!=1

2109 0
2110   function Se(e){var n,i=typeof e,o="object"==i,a=o||"function"==i
2111   if(Ae(t))return a?Pe(e,t):void 0
2112   void 0===(n=e[t])&&(o!=0||t in e)||"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113   if(&&ee()){Z(i,e,t);(Array.isArray(n)||(!o.r.isEmberArray)(n))&&Z(i,e,n,"[]");(!o.r.isProxy)(n)&&Z(i,e,n,"content"))
2114   n=S(e,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115   var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116   ●if(null!=a)●return Ne(a,o,n) o = "(1) LinkedIn", n = "(1) LinkedIn"
2117   if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+" could not be found.')})(e,t,n,i) i = undefined, e
2118   var o,a=○(o,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119   if(void 0==s&&q.has(s)){e[t]=n = "(1) LinkedIn"
2120   return n}if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121   o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122   return n}}function xe(){var Me=(function(e){(o,t.inheritsLoose)(i,e)
2123   function i(t){var n;(n=e.call(this)||this).__volatile__=1
2124   n. readOnly!=1

```

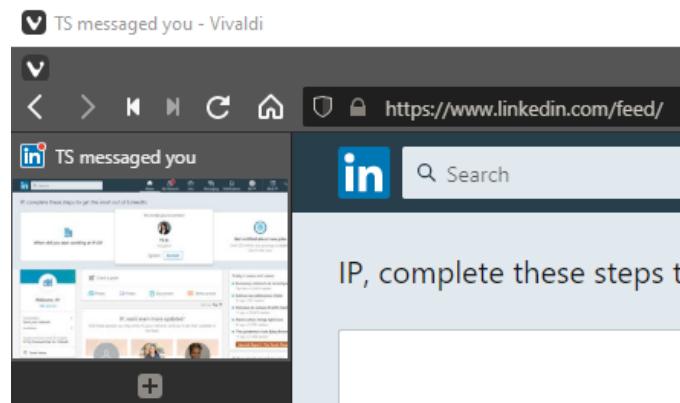
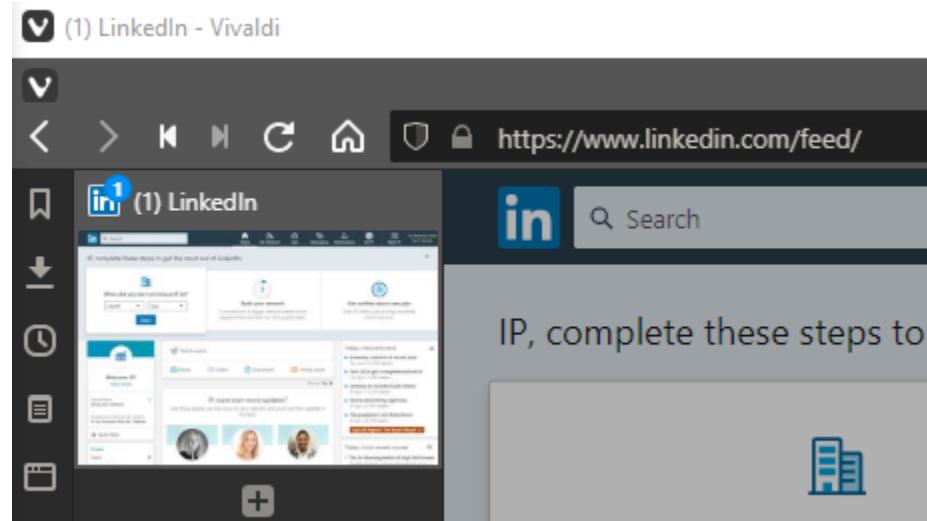


See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>



See, e.g., LinkedIn User Notification page at <https://www.linkedin.com/notifications>

215. The LinkedIn Application generates an event notification for the event. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which generates a notification to indicate to the user that the user has received a message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("_loopTitles","",concat("document-title-poller","_").concat(Date.now())))
},resetTitle:function(){if(t.default&&this._titles.length>0){this._titles.pop()
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)})
23413 e.default=n}
23414 define("image-edit-base/components/bg-image-edit", ["exports", "image-edit-base/components/picturecropper", "image-edit

```

```

2112 void o===(n=e[t])&&(o||t in e){ function i=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&ee()){Z ie(e,t);(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z ie(n,[ ]);(0,r.isProxy)(n)&&Z ie(n,"content"))
2114 n=S(e,n,[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 ● if(null!=a)●return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+' could not be found.')})(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t))else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n. readOnly=!1

```

```

2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"==i,a=o||"function"==i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0===(n=e[t])&&(o||t in e){"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&ee()){Z ie(e,t);(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z ie(n,[ ]);(0,r.isProxy)(n)&&Z ie(n,"content"))return n}functi
2114 n=S(e,n,[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 ● if(null!=a)●return Ne(a,o,n)o = "LinkedIn", n = "LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+' could not be found.')})(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!=n&&le(e,t))else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n. readOnly=!1

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

216. The LinkedIn Application associates the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which inputs the created notification into an array of title strings and inputs the created notification into an array of title strings which are specifically for use by the user's browser to display in the titlebar of the user's browser which is displayed on the display of the user's device:

```

define("html-document-title/services/document-title-poller", ["exports", "use strict"]
Object.defineProperty(e, "__esModule", { value: !0 })
e.default = void 0
var n = Ember.Service.extend({
  init: function () {
    this._super.apply(this, arguments)
    this._titles = []
    this._loopNumber = 0
    this._defaultDocumentTitle = ""
  },
  addTitle: function (e) {
    if (!this._titles.length && t.default) {
      this.updateDefaultTitle(document.title)
      this._titles = [e, this._defaultDocumentTitle]
    } else {
      this._titles.unshift(e)
      this._titles = this._titles.unique()
      if (!this._pollerEnabled) {
        this._pollerEnabled = !0
        this.pollTask("_loopTitles", "" . concat("document.title"))
      }
    }
  }
})

```

```

8415 define("ember-lifeline/poll-task", ["exports", "ember-lifeline/utils/get-task", "ember-lifeline/utils/disposable"],
8416   "use strict"
8417   Object.defineProperty(e, "__esModule", {
8418     value: !0
8419   })
8420   e._setRegisteredPollers = function(e) {
8421     a = e
8422   }
8423   e.setShouldPoll = function(e) {
8424     o = e
8425   }
8426   e.pollTask = function(e, r) {
8427     var c, m = arguments.length > 2 && void 0 !== arguments[2] ? arguments[2] : 1(), u = (0,
8428       t.default)(e, r, "pollTask"), d = function() {
8429       return u.call(e, c)
8430     }, p = a.get(e)
8431     if (!p) {
8432       p = new Set
8433       a.set(e, p);
8434       (0,
8435         n.registerDisposable)(e, (function(e, t) {
8436           return function() {
8437             t.forEach((function(t) {
8438               s(e, t)
8439             })
8440           )
8441         }
8442       })(e, p))
8443     }
8444     p.add(m)
8445     c = (function() {
8446       if (o)
8447         return o()
8448       return !Ember.testing
8449     })() ? d : function() {
8450       i[m] = d
8451     }
8452   }
8453   u.call(e, c)
8454   return m
8455 }
8456 }
8457 e.cancelPoll = s
8458 e.queuedPollTasks = void 0
8459 var a = new WeakMap
8460 var o, r = 0
8461 var i = Object.create(null)
8462 e.queuedPollTasks = i
8463 function s(e, t) {
8464   var n
8465   if ("number" == typeof e || "string" == typeof e)
8466     n = e
8467   else {
8468     var o = a.get(e)
8469     n = t
8470     void 0 !== o && o.delete(n)
8471   }
8472   delete i[n]
8473 }
8474 function l() {
8475   return r++
8476 }
8477 })

```

```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("_loopTitles","".concat("document-title-poller","_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)})
23413 e.default=n}
23414 define("image-edit-base/components/bc-image-edit","exports","image-edit-base/components/picture-converter","image-edit

```

```

2112 void o===(n-e[t])&&(!t in e){function i=typeof e.unknownProperty||(n.e.unknownProperty(t))
2113 if(a&&ee()){Z(ie(e,t));(Array.isArray(n)||(!r.isEmberArray)(n))&&Z(ie(n,"[]"));(!r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=S(e,n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split(".");
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null==a){return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new L.default('Property set failed: object in path "'+i.join("")+'" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!=s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t)else e.setUnknownProperty(t,n)
2122 return n}function xe(t){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))|this).volatile=!1
2124 n.readOnly=!1

```

```
8600 Id(this,t)
8601 Ud("tooltip: must be passed a control",n)
8602 var a=Id(this,(t.__proto__||Object.getPrototypeOf(t)).apply(this,arguments))
8603 a._player=e
8604 a._control=n
8605 a._titleText=a._control.el().getAttribute("title")
8606 a.setText(r)
8607 a.on(n,"mouseenter",a._onMouseenter)
8608 a.on(n,"mouseleave",a._onMouseleave)
8609 a.on(n,"dispose",function(){return a.dispose()})
8610 return a|At(t,[{key:"createEl",value:function(){var e=Ip[this.options._side]||"
8611 return xd(t.prototype.__proto__||Object.getPrototypeOf(t.prototype),"createEl",this).call(this,"div",{className:"vjs-tooltip "+e},{"
8612 this._control.el().removeAttribute("title")
8613 this._isShowing!=0}}],[{key:"hide",value:function(){if(this._isShowing){this.removeClass("vjs-tooltip-active")}
8614 this._control.el().setAttribute("title",this._titleText)
8615 this._isShowing!=1}}],[{key:"setText",value:function(e){var t=this.player.localize(e)
8616 Ep.textContent(this.el(),t)}},{key:"_onMouseenter",value:function(){this.show()}},{key:"_onMouseleave",value:function(){this.hide()}}
8617 return t}])
8618 An prototype options ={name:"ControlTooltip"}
```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

217. The LinkedIn Application provides the at least one of the plurality of character strings in the title array to a process executed by a processor. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the entries in the titlebar array for use by the code on the user's browser which is run by the processor of the user's device:

# An Introduction to JavaScript

Let's see what's so special about JavaScript, what we can achieve with it, and which other technologies play well with it.

## What is JavaScript?

*JavaScript* was initially created to “make web pages alive”.

The programs in this language are called *scripts*. They can be written right in a web page's HTML and run automatically as the page loads.

Scripts are provided and executed as plain text. They don't need special preparation or compilation to run.

In this aspect, JavaScript is very different from another language called Java.

Today, JavaScript can execute not only in the browser, but also on the server, or actually on any device that has a special program called the [JavaScript engine](#).

The browser has an embedded engine sometimes called a "JavaScript virtual machine".

Different engines have different "codenames". For example:

- [V8](#) – in Chrome and Opera.
- [SpiderMonkey](#) – in Firefox.
- ...There are other codenames like "Trident" and "Chakra" for different versions of IE, "ChakraCore" for Microsoft Edge, "Nitro" and "SquirrelFish" for Safari, etc.

The terms above are good to remember because they are used in developer articles on the internet. We'll use them too. For instance, if "a feature X is supported by V8", then it probably works in Chrome and Opera.

#### How do engines work?

Engines are complicated. But the basics are easy.

1. The engine (embedded if it's a browser) reads ("parses") the script.
2. Then it converts ("compiles") the script to the machine language.
3. And then the machine code runs, pretty fast.

The engine applies optimizations at each step of the process. It even watches the compiled script as it runs, analyzes the data that flows through it, and further optimizes the machine code based on that knowledge.

## What can in-browser JavaScript do?

Modern JavaScript is a "safe" programming language. It does not provide low-level access to memory or CPU, because it was initially created for browsers which do not require it.

JavaScript's capabilities greatly depend on the environment it's running in. For instance, [Node.js](#) supports functions that allow JavaScript to read/write arbitrary files, perform network requests, etc.

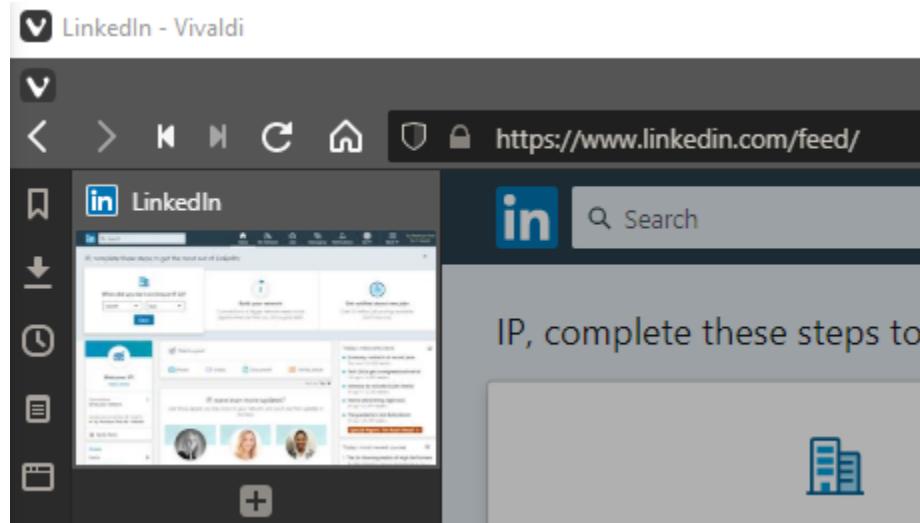
In-browser JavaScript can do everything related to webpage manipulation, interaction with the user, and the webserver.

For instance, in-browser JavaScript is able to:

- Add new HTML to the page, change the existing content, modify styles.
- React to user actions, run on mouse clicks, pointer movements, key presses.
- Send requests over the network to remote servers, download and upload files (so-called [AJAX](#) and [COMET](#) technologies).
- Get and set cookies, ask questions to the visitor, show messages.
- Remember the data on the client-side ("local storage").

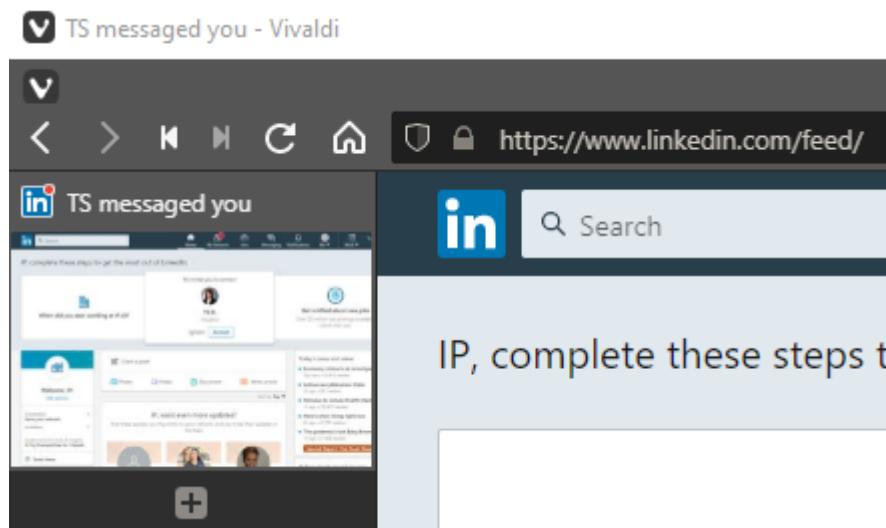
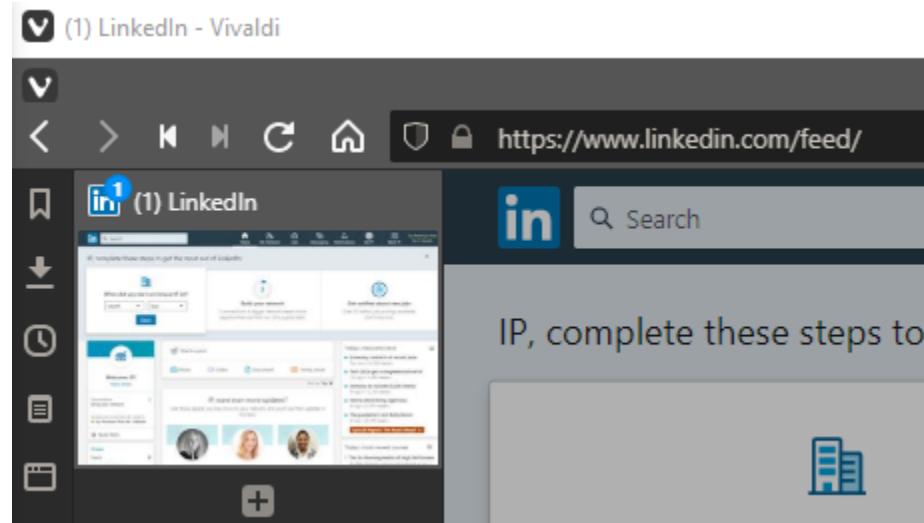
See, e.g., An Introduction to JavaScript on [JavaScript.info at https://JavaScript.info/intro](https://JavaScript.info/intro)

218. The LinkedIn Application provides an alternative title based on the at least one of the plurality of character strings to the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the user's browser with titles for each of the entries in the titlebar, which is sent to the processor of the user's device for updating the HTML run and displayed by the user's browser:



```
23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)}  
23398 this._titles=[]  
23399 this._loopNumber=0  
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc  
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)  
23402 this._titles=this._titles.uniq()  
23403 if(!this._pollerEnabled){this._pollerEnabled=!0  
23404 this.pollTask("_loopTitles","_.concat("document-title-poller","_").concat(Date.now()))}}},resetTitle:function(){if(t  
23405 this._loopNumber=0  
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle  
23407 this._pollerEnabled!=1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r  
23408 this._titles[n-1]=this._defaultDocumentTitle  
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"  
23410 t.default&&a&&(document.title=a)  
23411 this._loopNumber++  
23412 this._pollerEnabled&&this.runTask(e,1500)})}  
23413 e.default=n})  
23414 define("image-edit-base/components/hg-img-edit", ["exports", "image-edit-base/components/picture-cropper", "image-edit
```

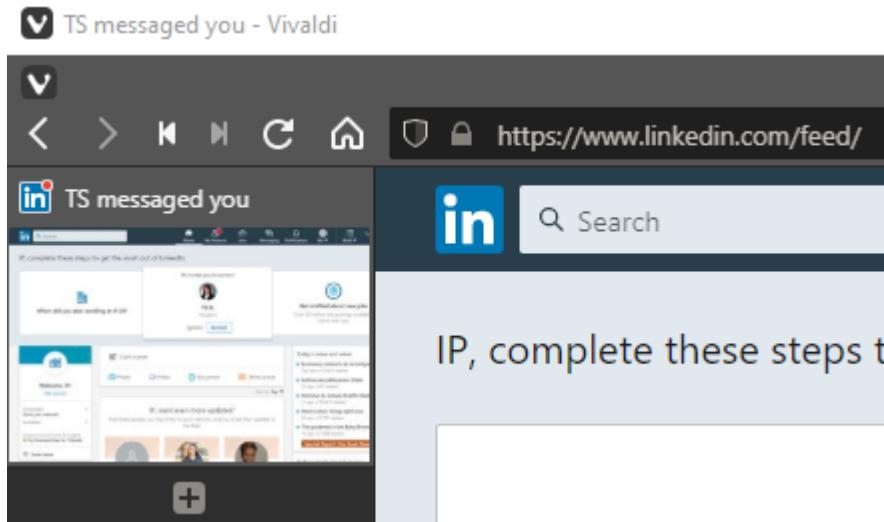
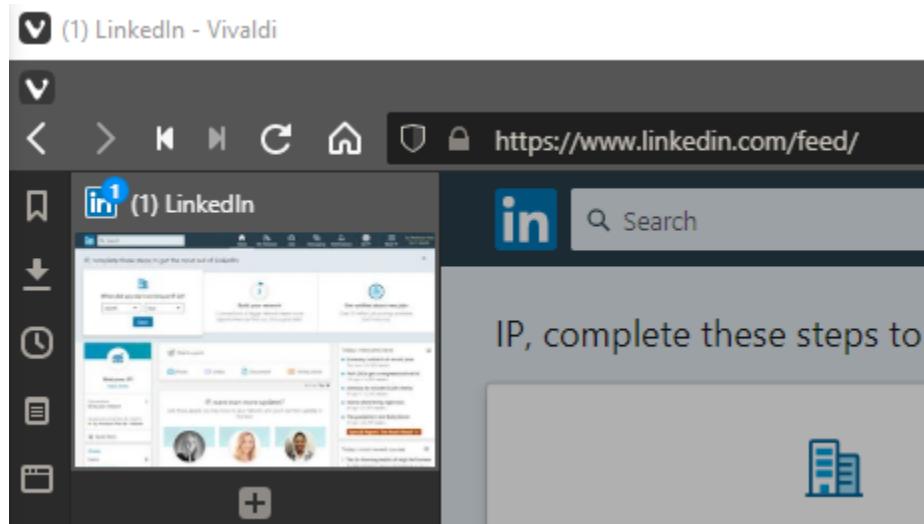
```
1112 void 0===(n=e[t])&&(t in e){function i=typeof e.unknownProperty||(n=e.unknownProperty(t))  
1113 if(a&&e()){z=(ie,e,t);(Array.isArray(n)||(!o,r.isEmberArray)(n))&&z.ie(n,"[]"));(o,r.isProxy)(n)&&z.ie(n,"content"))  
1114 n=S(e,n,r[i]);return n}function Ne(e,t,n,i){if(e.isDestroyed){if(A(e))return (function(e,t,n,r){var i=t.split("."),  
1115 var a=P(e,i);a={enumerable: true, configurable: true, get: f, set: f}  
1116 o||f(null)==a}||return Ne(a,o,n))}o="LinkedIn", n="(1) LinkedIn"  
1117 if(r)throw new l.default('Property set failed: object in path "'+i.join(".")+'"' could not be found.')})(e,t,n,i);i  
1118 var o,a=(o,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set;o="LinkedIn", a={enumerable: true, configurable: tr  
1119 if(void 0!==s&&q.has(s)){e[t]=n;n="(1) LinkedIn"  
1120 return n}if(void 0!=o=(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n  
1121 o!=null&&e(t);else e.setUnknownProperty(t,n)  
1122 return n}};function xe(){var Me=(function(e){(o,t.inheritsLoose)(i,e)  
1123 function i(t){var n;(n=e.call(this)||this)._volatile=!1  
1124 n._readOnlyv=11
```



See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed/>

219. The LinkedIn Application uses the alternative title as a title in association with the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which causes the user's browser to use the alternative title or titles as the titlebar for the user's browser by updating the HTML run and displayed by the user's

browser:



```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("loopTitles","").concat("document-title-poller","_").concat(Date.now()),resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=1},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)})
23413 e.default=n)
23414 define("image-edit-base/components/bg-image-edit",["exports","image-edit-base/components/picture-cropper","image-edit

```

```

2112 void o===(n=e[t])&&(!o||t in e){function i=typeof e.unknownProperty||!(n=e.unknownProperty(t))
2113 if(a&&ee()){Z ie(e,t);(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z ie(n,"[]");(!0,r.isProxy)(n)&&Z ie(n,"content")
2114 n=Se(n,r[i]);return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t)){return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a){return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+' could not be found.')}(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&l(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){{!0,t.inheritsLoose})(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n.readOnly=!1

```

```

2109 o
2110 function Se(e,t){var n,i=typeof e,o="object"==i,a=o||"function"==i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0===(n=e[t])&&(!o||t in e){function i=typeof e.unknownProperty||!(n=e.unknownProperty(t))
2113 if(a&&ee()){Z ie(e,t);(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z ie(n,"[]");(!0,r.isProxy)(n)&&Z ie(n,"content")})return n}functi
2114 n=Se(n,r[i]);return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t)){return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a){return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+' could not be found.')}(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0==s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!=n&&l(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){{!0,t.inheritsLoose})(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n.readOnly=!1

```

```

1890 c=(function(){if(
1891 return!Ember.test
1892 u.call(e,_)
1893 return m}
1894 e.cancelPolls=
1895 e.queuedPollTasks
1896 var a=new WeakMap
1897 var o,r=0
1898 var i=Object.create
1899 e.queuedPollTasks
1900 function s(e,t){v
1901 if("number"==typeof
1902 else{var o=a.get(
1903 n=t
1904 void 0!=o&&o.dele
1905 define("ember-lifeline",["use
1906 Object.defineProperty
1907 e._setRegisteredTimelin
1908 e.runTask=function(t){var o=arguments.length>2&&void 0==arguments[2]?arguments[2]:0
1909 if(e.isDestroying){return a
1910 var r=(0,n.default)(e,t,"runTask"),s=i(e),l=Ember.run.later((function(){s.delete(l)
1911 r.call(e)),o)
1912 s.add(l)
1913 s

```

```

2101 u||1==c?Object.defineProperty(e,t,{configurable:!0,enumerable:r,writable:!0,value:l}):e[t]=l}else{l=r
2102 Object.defineProperty(e,t,r)o.isPrototypeOfMeta
2103 t.lastRevision=(0,s.value)(t.tag)))
2104 A.has(e)&&A.get(e).forEach(function(t){t.tag=t.lastRevision=(0,s.value)(t.tag)}))(e)
2105 t.lastRevision=(0,s.value)(t.tag))))(e)
2106 "function"==typeof e.didDefineProperty&&e.didDe
2107 function Ae(e){return"string"==typeof e&&-1==e
2108 e.PROXY_CONTENT=Ce
2109 o
2110 function Se(e,t){var n,i=typeof e,o="object"==
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0===(n=e[t])&&(!o||t in e||"function"!=ty
2113 if(a&&ee()){Z ie(e,t);(Array.isArray(n)||(!0,r
2114 n=Se(n,r[i]);return n}function Ne(e,t,n,i){if(
2115 var a=Pe(e,i) a = {enumerable: true, configura
2116 if(null!=a){return Ne(a,o,n)o = "(1) LinkedIn
2117 if(!r)throw new l.default('Property set failed:
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a
2119 if(void 0==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&l(e,t)}else e.setUnknownProperty(t,n)

```

```

2101 u||!i:object.defineProperty(e,t,{configurable:!0,enumerable:c,writable:!0,value:i}):e[t]=i}else{l=r
2102 Object.defineProperty(e,t,r):isPrototypeOfMeta(e)||((function(e){C.has(e)&&C.get(e).forEach((function(t){t.tag=(0,s.combine)(we,e,t.p
2103 t.lastRevision=(0,s.value)(t.tag)))
2104 A.has(e)&&A.get(e).forEach((function(t){t.tag=(0,s.combine)(we,e,t.path)))
2105 t.lastRevision=(0,s.value)(t.tag)}))}(e)
2106 "function"==typeof e.didDefineProperty&&e.didDefineProperty(e,t,1){var ke=new r.Cache(1e3,function(e){return e.indexOf(".")})
2107 function Ae(e){return"string"==typeof e&&-1==ke.get(e)}var Ce=(0,r.symbol)("PROXY_CONTENT")
2108 e.PROXY_CONTENT=Ce
2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"==i,a=o|"function"==i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(i=t["function"]!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&ee){{n=ie(e,t)};(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))}return n}functi
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e =
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a) return Ne(a,o,n) o = "(1) LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+' could not be found.')(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?vo "title" : o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!=s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e||!i in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&f(e,t)}else e.setUnknownProperty(t,n)

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

220. Defendant has directly infringed, and continues to directly infringe, the claims of the ‘135 Patent, including at least those noted above, including by at least making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a).

221. LinkedIn has had at least constructive notice of the ‘135 Patent since at least its issuance. LinkedIn will have been on actual notice of the ‘135 Patent since, at the latest, the service of this complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the ‘135 Patent.

222. EBT believes and contends that, at minimum, LinkedIn’s knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the ‘135 Patent since receiving notice of its infringement of the ‘135 Patent, is necessarily willful, wanton, malicious, in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a finding of willful infringement. Accordingly, since at least receiving notice of this suit, LinkedIn has willfully infringed the ‘135 Patent.

#### **COUNT IV – INFRINGEMENT OF U.S. PATENT No. 8,402,179**

223. Plaintiff is the owner of the ‘179 Patent and has all substantial rights to the ‘179 Patent, including the right and standing to sue and recover damages for past, present, and future

infringement of the ‘179 patent.

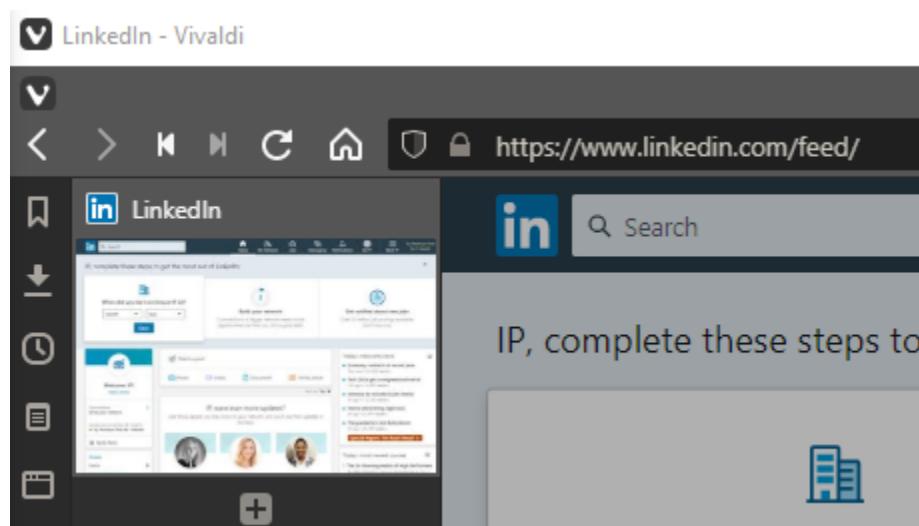
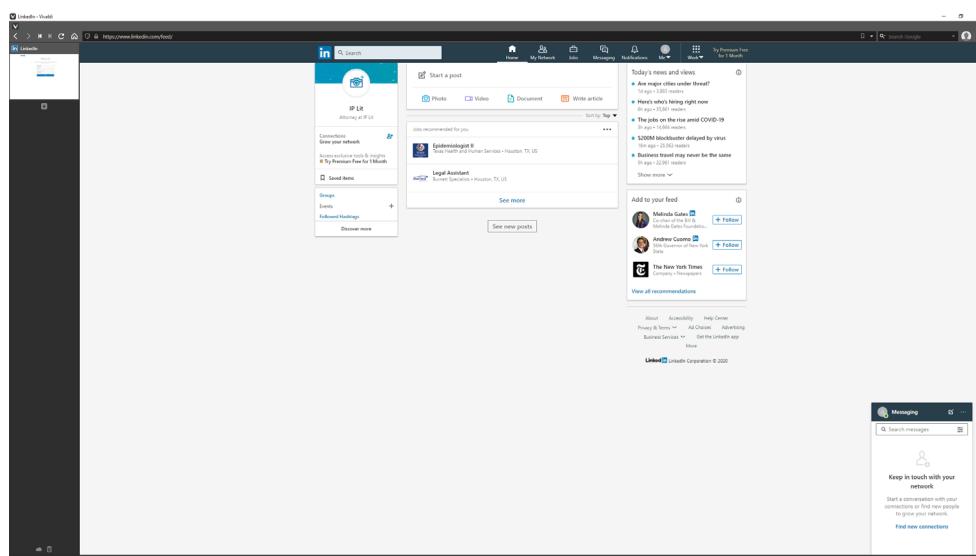
224. Claim 1 of the ‘179 Patent covers a method comprising “processing an event that calls for user notification; generating an event notification for the event; storing the event notification in an array; providing the event notification from the array to a process executed by a processor; using the event notification as a title in association with the process; providing an alternative title from the array to the process; using the alternative title as a title in association with the process.”

225. Defendant has infringed, and is now infringing, the ‘179 Patent, including at least claim 1, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising the practicing of methods and/or providing of systems, without authority from Plaintiff, for notifying a user of the occurrence of events and notification apparatuses and functionality, including notifying a user of the occurrence of an event by modification of the user’s browser title bar based on the specific event that has occurred, including as claimed in the ‘179 asserted claims. On information and belief, Defendant practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising [www.linkedin.com](http://www.linkedin.com).

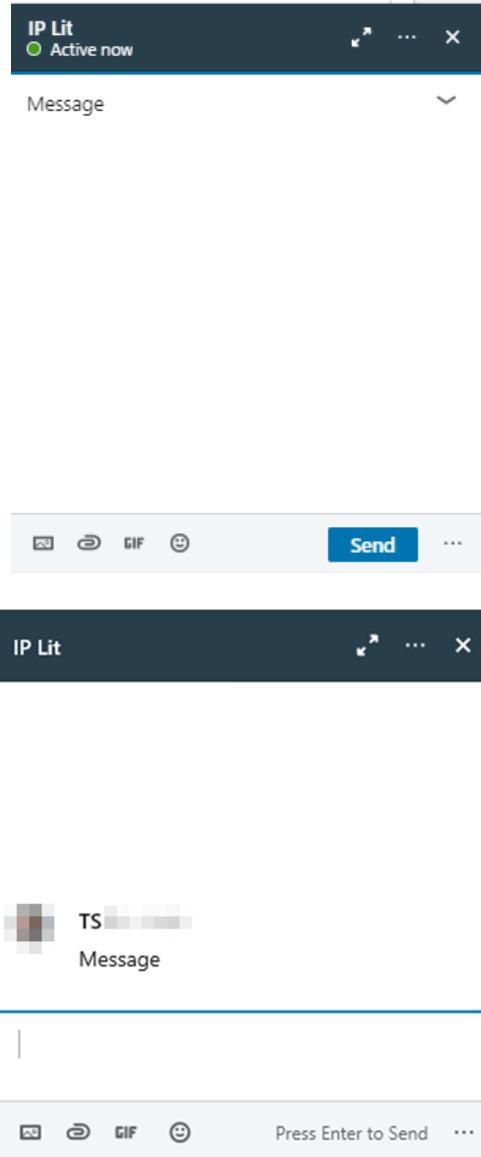
226. Without limitation, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods comprises processing an event that calls for user notification; generating an event notification for the event; storing the event notification in an array; providing the event notification from the array to a process executed by a processor; using the event notification as a title in association with the process; providing an alternative title from the array to the process; using the alternative title as a title in association with the process. For example, the LinkedIn Application system permits a user’s device to notify the user, via the user’s browser, that the user has notifications for an event, such as a new

message and/or other posting on the LinkedIn website, including via updating, modifying, and/or otherwise altering the title bar of the user's browser, including instructing the user's browser to receive notifications regarding messages the user has received from other LinkedIn users and/or posting of other LinkedIn users.

227. The LinkedIn Application processes an event that calls for user notification. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which processes code or other instructions received from LinkedIn that the user has received a message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>



See, e.g., Second LinkedIn User Feed page at <https://www.linkedin.com/feed>

```

2113 if(!o.access()){e(e(c,c)),(Array.isArray(n)||!(e.r.isIMbededInFeed)(n))&&g(e(n,[])),(o,r.isProxy)(n)&&z(e(n, content ))
2114 n=S(e,n,r[i])}return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {value: "/voyager/api/messaging/conversations/6694727273901...4JGdp_RcmK3lp_RcLnVybjpsa1pmYWJyaM6
2116 ● if(null!=a)● return ●Ne(a,o,n) o = "/voyager/api/messaging/conversations/6694727273901359104/events?q=syncToken&re
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i
2118 var o,a=●(0,r.lookupDescriptor)●(e,t),s=●null==a?void 0:a.set o = "/voyager/api/messaging/conversations/6694727
2119 ● if(void 0!==s&&q.●has(s)){●e[t]=r n = "/voyager/api/messaging/conversations?keyVersion=LEGACY_INBOX&q=syncToken&s
2120 ● return n}● if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){●e[t]=n
2121 o=n&&e[e[t]]else e.setUnknownProperty(t,n)
2122
2123 if(!o.access()){e(e(c,c)),(Array.isArray(n)||!(e.r.isIMbededInFeed)(n))&&g(e(n,[])),(o,r.isProxy)(n)&&z(e(n, content ))
2124 n=S(e,n,r[i])}return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2125 var a=Pe(e,i) a = {value: "urn:li:fs_event:(6694727273901359104,1694772476838248448_500)", writable: true, enumera
2126 ● if(null!=a)● return ●Ne(a,o,n) o = undefined, n = "urn:li:fs_event:(6694727273901359104,I6694772476838248448_500)"
2127 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i
2128 var o,a=●(0,r.lookupDescriptor)●(e,t),s=●null==a?void 0:a.set o = undefined, a = {value: "urn:li:fs_event:(6694
2129 ● if(void 0!==s&&q.●has(s)){●e[t]=r n = "urn:li:fs_event:(6694727273901359104,I6694772476838248448_500)"
2130 ● return n}● if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){●e[t]=n
2131 o=n&&e[e[t]]else e.setUnknownProperty(t,n)

```

```

1835 e.isBlank=He
1836 e.isPresent=function(e){return!He(e)}
1837 e.beginPropertyChanges=ce
1838 e.changeProperties=he
1839 e.endPropertyChanges=fe
1840 e.notifyPropertyChange=le
1841 e.defineProperty=Te
1842 e.isElementDescriptor=U
1843 e.nativeDescDecorator=V
1844 e.descriptorForDecorator=D
1845 e.descriptorForProperty=I
1846 e.isClassicDecorator=B
1847 e.setClassicDecorator=F
1848 e.getChainTagsForKey=we
1849 e.getProperties=function(e,t){var n={},r=arguments,i=1
1850 if(2==arguments.length&Array.isArray(t)){i=0
1851 r=arguments[1]}for(;i<r.length;i++)n[r[i]]=Se(e,r[i])
1852 return n}
1853 e.setProperty=function(e,t){if(null==t||"object"!=typeof t) return t
1854 he((function(){for(var n,r=Object.keys(t),i=0;i<r.length;i++){n[t[i]]=e,r[i]=n}}))
1855 Ne(e,n,t[n])})
1856 return t}
1857 e.expandProperties=0e

```

```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&t.default){this.updateDefaultTitle(doc
23401 this._titles[e,this._defaultDocumentTitle]]else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.runTask("loopTitles","_".concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&a&&document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)})
23413 e.default=n}
23414 define("image-edit-base/components/base-image-edit","exports","image-edit-base/components/picture-cropper","image-edit

```

```

2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116 ● if(null!=a)● return Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117 if(!r)throw new l.default("Property set failed: object in path '"+i.join(".")+"' could not be found.'))(e,t,n,i) i
2118 var o,a=●(0,r.lookupDescriptor("latestEvent") s1==a:void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119 ● if(void 0!==s&&q.●has(s)){● o=j=n = u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120 ● return n}● if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){● e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)

```

```

2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116 ● if(null!=a)● return Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117 if(!r)throw new l.default("Property set failed: object in path '"+i.join(".")+"' could not be found.'))(e,t,n,i) i
2118 var o,a=●(0,r.lookupDescriptor("event") s=●null==a:void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119 ● if(void 0!==s&&q.●has(s)){● o=j=n = u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120 ● return n}● if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){● e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)

```

```

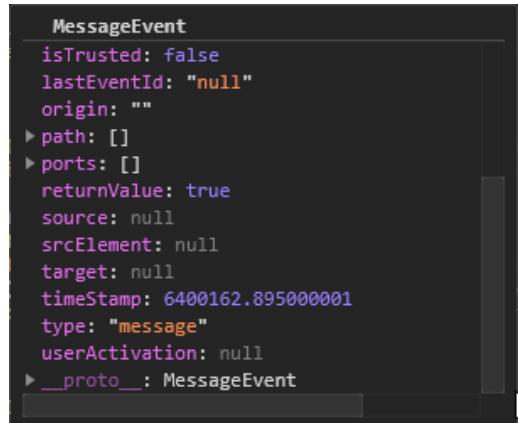
2112 void 0===(n=e[t])&&(t in e){function i=typeof e.unknownProperty||(0,e.unknownProperty(t))
2113 if(a&&ee())(Z(i(e,t));(Array.isArray(n)||(!r.isEmberArray)(n))&&Z(i(e,n,"["]))||(!r.isProxy)(n)&&Z(i(e,n,"content")))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 ● if(null!=a)● return Ne(a,o,n) o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default("Property set failed: object in path '"+i.join(".")+"' could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a:void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!=s&&q.●has(s)){● o=j=n = "LinkedIn"
2120 return n}● if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){● e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n.readOnly=!1

```

```

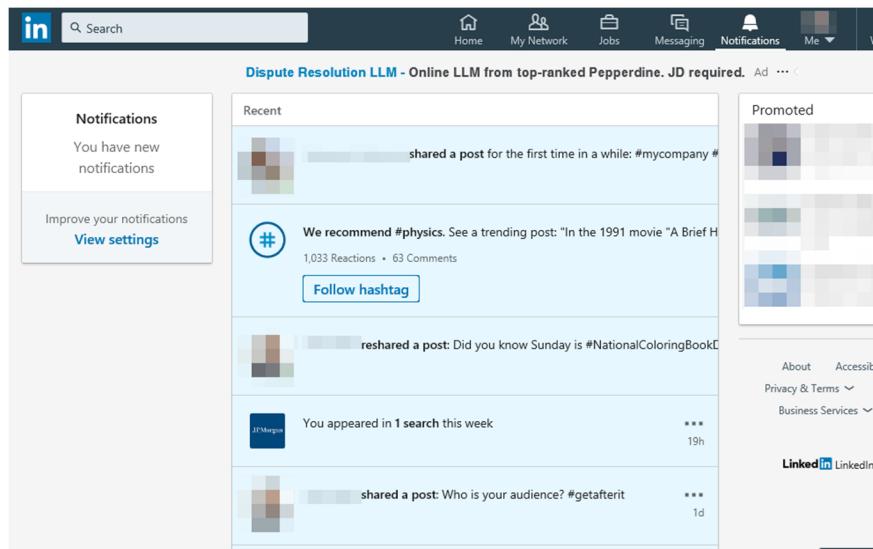
2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"==i,a=o||"function"==i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0===(n=e[t])&&(t in e){function i=typeof e.unknownProperty||(0,e.unknownProperty(t))
2113 if(a&&ee())(Z(i(e,t));(Array.isArray(n)||(!r.isEmberArray)(n))&&Z(i(e,n,"["]))||(!r.isProxy)(n)&&Z(i(e,n,"content")))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e = i
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 ● if(null!=a)● return Ne(a,o,n) o = "(1) LinkedIn", n = "LinkedIn"
2117 if(!r)throw new l.default("Property set failed: object in path '"+i.join(".")+"' could not be found.'))(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a:void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!=s&&q.●has(s)){● o=j=n = "LinkedIn"
2120 return n}● if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){● e[t]=n o = "(1) LinkedIn"
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n.readOnly=!1

```



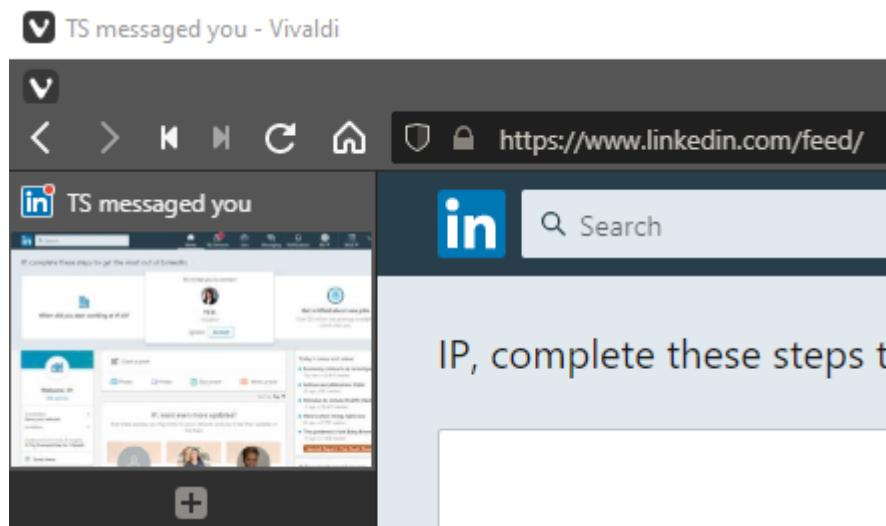
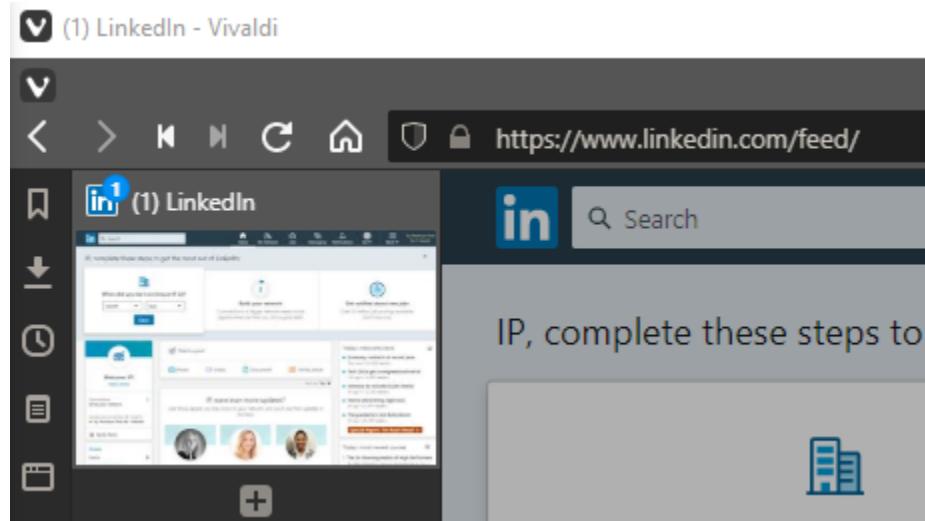
```
MessageEvent
isTrusted: false
lastEventId: "null"
origin: ""
path: []
ports: []
returnValue: true
source: null
srcElement: null
target: null
timeStamp: 6400162.895000001
type: "message"
userActivation: null
__proto__: MessageEvent
```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>



See, e.g., LinkedIn User Notification page at <https://www.linkedin.com/notifications>

228. The LinkedIn Application generates an event notification for the event. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which generates a notification to indicate to the user that the user has received a message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398   this._titles=[]
23399   this._loopNumber=0
23400   this._pollerEnabled=false
23401   this._documentTitle=""
23402   addTitle=function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23403   this._titles=[e,t.defaultDocumentTitle])else this._titles.unshift(e)
23404   this._titles=this._titles.uniq()
23405   if(this._pollerEnabled){this._pollerEnabled=t
23406   this._loopTask="_loopTitles"+_.concat("document-title-poller","_").concat(Date.now()))
23407   resetTitle=function(){if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23408   this._pollerEnabled=t}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23409   var a=this._titles[this._loopNumber]
23410   if(a){document.title=a}
23411   a="TS messaged you"
23412   this._loopNumber++
23413   this._pollerEnabled&&this.runTask(e,1500)}}
23414   e.default=n
23415   define("image-edit-base/components/bg-img-edit",{"exports": "image-edit-base/components/picture-scanner", "image-edit"

```

```

2112 void o===(n=e[t])&&(o||t in e){ function i=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&ee()){Z ie(e,t));(Array.isArray(n)||!(o.r.isEmberArray)(n))&&Z ie(n,"[ ]);(o.r.isProxy)(n)&&Z ie(n,"content"))
2114 n=S(e,n,r[i]);return n}function Ne(e,t,n,i){if(!e(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 ●if(null!=a)●return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i
2118 var o,a=(o.r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!=s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(o.t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this).volatile=!1
2124 n. readOnly=!1

```

```

2109 o
2110 function Se(e,t){var n,i=typeof e,o="object"==i,a=o||"function"==i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(o||t in e||"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&ee()){Z ie(e,t));(Array.isArray(n)||!(o.r.isEmberArray)(n))&&Z ie(n,"[ ]);(o.r.isProxy)(n)&&Z ie(n,"content"))return n}functi
2114 n=S(e,n,r[i]);return n}function Ne(e,t,n,i){if(!e(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 ●if(null!=a)●return Ne(a,o,n)o = "LinkedIn", n = "LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i = undefined, e
2118 var o,a=(o.r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!=s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n}if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(o.t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this).volatile=!1
2124

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

229. The LinkedIn Application stores the event notification in an array. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which inputs the created notification into an array of title strings:

```

define("html-document-title/services/document-title-poller", ["exports", 'use strict'];
Object.defineProperty(e, "__esModule", { value: !0 });
e.default = void 0;
var n = Ember.Service.extend({
  init: function () {
    this._super.apply(this, arguments);
    this._titles = [];
    this._loopNumber = 0;
    this._defaultDocumentTitle = "";
  },
  addTitle: function (e) {
    if (!this._titles.length && t.default) {
      this.updateDefaultTitle(document.title);
      this._titles = [e, this._defaultDocumentTitle];
    } else {
      this._titles.unshift(e);
      this._titles = this._titles.uniq();
      if (!this._pollerEnabled) {
        this._pollerEnabled = !0;
        this.pollTask("_loopTitles", "" . concat("document.title"));
      }
    }
  }
});

```

```

6410 define("ember-lifeline/poll-task", ["exports", "ember-lifeline/utils/get-task", "ember-lifeline/utils/disposable"], {
6411   "use strict"
6412   Object.defineProperty(e, "__esModule", {
6413     value: !0
6414   })
6415   e._setRegisteredPollers = function(e) {
6416     a = e
6417   }
6418   e.setShouldPoll = function(e) {
6419     o = e
6420   }
6421   e.pollTask = function(e, r) {
6422     var a = arguments.length > 2 ? arguments[2] : arguments[2] : 1(), u = (0,
6423       t, r, "pollTask"), d = Function() {
6424       return u.call(e, c)
6425     }, p = a.get(e)
6426     if (p) {
6427       p = new Set
6428       p.add(e, p)
6429       (0,
6430         n.registerDisposable)(e, (function(e, t) {
6431           return function() {
6432             t.forEach((function(t) {
6433               s(e, t)
6434             })
6435           })
6436         })(e, p))
6437       p.add(m)
6438       c = (function() {
6439         if (o)
6440           return o()
6441         else
6442           return Ember.testing
6443         }())
6444       if (d)
6445         i[m] = d
6446       u.call(e, c)
6447       return m
6448     }
6449     e.cancelPoll = s
6450     e.queuedPollTasks = void 0
6451     var a = new WeakMap
6452     var o = {}
6453     var d = Object.create(null)
6454     e.queuedPollTasks = i
6455     function s(e, t) {
6456       var n
6457       if ("number" === typeof e || "string" === typeof e)
6458         n = e
6459       else
6460         var o = a.get(e)
6461         n = t
6462         void 0 !== o && o.delete(n)
6463       i[e] = n
6464       delete i[n]
6465     }
6466     function t() {
6467       r++
6468     }
6469   }
6470 })
```

```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("_loopTitles","","concat("document-title-poller","_").concat(Date.now()))),resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=1},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){n
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)})
23413 e.default=n}
23414 define("image-edit-base/components/bc-img-edit","exports","image-edit-base/components/picture-cropper","image-edit
23415
23416
23417
23418
23419
23420
23421
23422
23423
23424
23425
23426
23427
23428
23429
23430
23431
23432
23433
23434
23435
23436
23437
23438
23439
23440
23441
23442
23443
23444
23445
23446
23447
23448
23449
23450
23451
23452
23453
23454
23455
23456
23457
23458
23459
23460
23461
23462
23463
23464
23465
23466
23467
23468
23469
23470
23471
23472
23473
23474
23475
23476
23477})})
```

```

2112 void o===(n-e[t])&&(o||t in e){function i=typeof e.unknownProperty||(n-e.unknownProperty(t))
2113 if(a&&ee())Z(i,e,t);(Array.isArray(n)||@.r.isEmberArray())&&Z(i,e,n,"[]");@.r.isProxy)(n)&&Z(i,e,n,"content")
2114 n=S(e,n,r[i]);return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(A(e,t))return (function(e,t,n,r){var i=t.split(".");
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 @if(null!=a){return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new Error("Property set failed: object in path '"+i.join(".")+" could not be found.'))(e,t,n,i) i
2118 var o,@=(@.r.lookupDescriptor)(e,t),s=null===@?void 0:@.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t]))||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=@&@{e,t})else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=Function(e){@.t.inheritsLoose}(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n.readOnly=!1
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2999
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3199
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3389
3390
3391
3392
3393
3394
3395
3396
3397
3397
3398
3399
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3489
3490
3491
3492
3493
3494
3495
3496
3497
3497
3498
3499
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3598
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3698
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3798
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3819
3820
3821
3822
3823
3824
3825
3826
3827
3828
3829
3829
3830
3831
3832
3833
3834
3835
3836
3837
3838
3839
3839
3840
3841
3842
3843
3844
3845
3846
3847
3848
3849
38
```

```

8600 id(this,t)
8601 ①d("tooltip: must be passed a control",n)
8602 var a=①d(this,(t.__proto__||Object.getPrototypeOf(t)).apply(this,arguments))
8603 a.player=e
8604 a._control=
8605 a._titleText=a._control.el().getAttribute("title")
8606 a.setText(r)
8607 a.on(n,"mouseenter",a._onMouseenter)
8608 a.on(n,"mouseleave",a._onMouseleave)
8609 a.on(n,"dispose",function(){return a.dispose()})
8610 return a)Ad(t,[{key:"createEl",value:function(){var e=Ip[this.options_.side]||"
8611 return xd(t.prototype.__proto__||Object.getPrototypeOf(t.prototype),"createEl",this).call(this,"div",{className:"vjs-tooltip "+e},{"
8612 this._control.el().removeAttribute("title")
8613 this._isShowing!=0}),{key:"hide",value:function(){if(this._isShowing){this.removeClass("vjs-tooltip-active")
8614 this._control.el().setAttribute("title",this._titleText)
8615 this._isShowing!=1}}},{}],{key:"_onMouseenter",value:function(){var t=this.player.localize(e)
8616 Ep.textContent(this.el(),t)},key:"_onMouseleave",value:function(){this.show()},key:"_onMouseleave",value:function(){this.hide()})
8617 return t}])
8618 A.prototype.options =f{name:"ControlTooltip"}

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

230. The LinkedIn Application provides the event notification from the array to a process executed by a processor. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the entries in the titlebar array for use by the code on the user's browser which is run by the processor of the user's device:

## An Introduction to JavaScript

Let's see what's so special about JavaScript, what we can achieve with it, and which other technologies play well with it.

### What is JavaScript?

JavaScript was initially created to "make web pages alive".

The programs in this language are called *scripts*. They can be written right in a web page's HTML and run automatically as the page loads.

Scripts are provided and executed as plain text. They don't need special preparation or compilation to run.

In this aspect, JavaScript is very different from another language called *Java*.

Today, JavaScript can execute not only in the browser, but also on the server, or actually on any device that has a special program called the *JavaScript engine*.

The browser has an embedded engine sometimes called a "JavaScript virtual machine".

Different engines have different "codenames". For example:

- **V8** – in Chrome and Opera.
- **SpiderMonkey** – in Firefox.
- ...There are other codenames like "Trident" and "Chakra" for different versions of IE, "ChakraCore" for Microsoft Edge, "Nitro" and "SquirrelFish" for Safari, etc.

The terms above are good to remember because they are used in developer articles on the internet. We'll use them too. For instance, if a feature X is supported by V8, then it probably works in Chrome and Opera.

#### How do engines work?

Engines are complicated. But the basics are easy.

1. The engine (embedded if it's a browser) reads ("parses") the script.
2. Then it converts ("compiles") the script to the machine language.
3. And then the machine code runs, pretty fast.

The engine applies optimizations at each step of the process. It even watches the compiled script as it runs, analyzes the data that flows through it, and further optimizes the machine code based on that knowledge.

## What can in-browser JavaScript do?

Modern JavaScript is a "safe" programming language. It does not provide low-level access to memory or CPU, because it was initially created for browsers which do not require it.

JavaScript's capabilities greatly depend on the environment it's running in. For instance, Node.js supports functions that allow JavaScript to read/write arbitrary files, perform network requests, etc.

In-browser JavaScript can do everything related to webpage manipulation, interaction with the user, and the webserver.

For instance, in-browser JavaScript is able to:

- Add new HTML to the page, change the existing content, modify styles.
- React to user actions, run on mouse clicks, pointer movements, key presses.
- Send requests over the network to remote servers, download and upload files (so-called **AJAX** and **COMET** technologies).
- Get and set cookies, ask questions to the visitor, show messages.
- Remember the data on the client-side ("local storage").

*See, e.g., An Introduction to JavaScript on JavaScript.info at <https://JavaScript.info/intro>*

231. The LinkedIn Application uses the event notification as a title in association with the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which inputs the created notification into an array of title strings for use by the user's browser to display in the titlebar of the user's browser, including with the HTML title tag processed by the user's browser:

```

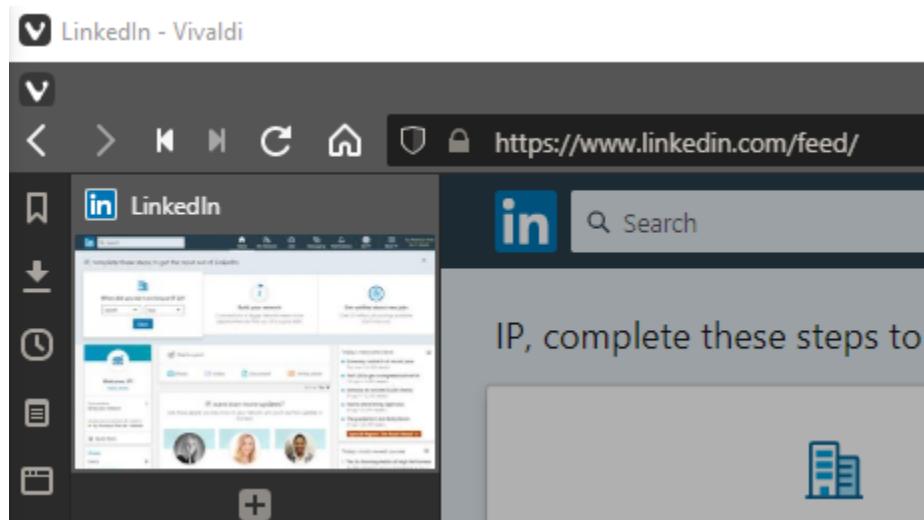
23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle=""
23401 this._titles=[e,this._defaultDocumentTitle]else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled!=0
23404 this.pollTask("_loopTitles","".concat("document-title-poller","_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled!=1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)})
23413 e.default=n}
23414 define("image-edit-base/components/bg-img-edit",["exports", "image-edit-base/components/picture-cropper", "image-edit
23415
23416 void o===(n=e[t])&&(t in e|| Function==typeof e.unknownProperty||(n=e.unknownProperty(t)))
23417 if(a&&ee())f(ie(e,t));(Array.isArray(n)||(@.r.isEmberArray)(n))&&Z(ie(n,"[ ])");(@.r.isProxy)(n)&&Z(ie(n,"content"))
23418 n=S(e,n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
23419 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
23420 if(null!=a)● return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
23421 if(!r)throw new l.default("Property set failed: object in path "+i.join(".")+" could not be found.")}(e,t,n,i) i
23422 var o,a=@.r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
23423 if(void 0!=s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
23424 return n}if(void 0==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
23425 o!=n&&le(e,t)else e.setUnknownProperty(t,n)
23426 return n}function xe(){var Me=(function(e){@.t.inheritsLoose)(i,e)
23427 function i(t){var n=(n=e.call(this))||this._volatile!=1
23428 n. readOnly=!1

```

```
8600 id(this,t)
8601 Ud("tooltip: must be passed a control",n)
8602 var a=Ld(this,(t.__proto__||Object.getPrototypeOf(t)).apply(this,arguments))
8603 a.player=e
8604 a._control=n
8605 a[_titleText=a._control.el().getAttribute("title")]
8606 a.setText(r)
8607 a.on(n,"mouseenter",a._onMouseenter)
8608 a.on(n,"mouseleave",a._onMouseleave)
8609 a.on(n,"dispose",function(){return a.dispose()}))
8610 return a)Ad(t,[{key:"createEl",value:function(){var e=Ip[this.options_.side]||"
8611 return xd(t.prototype.__proto__||Object.getPrototypeOf(t.prototype),"createEl",this).call(this,"div",{className:"vjs-tooltip "+e},{"
8612 this._control.el().removeAttribute("title")
8613 this._isShowing!=0}],{key:"hide",value:function(){if(this._isShowing){this.removeClass("vjs-tooltip-active")}
8614 this._control.el().setAttribute("title",this._titleText)
8615 this._isShowing!=1}],{key:"setText",value:function(e){var t=this.player.localize(e)
8616 Ep.textContent(this.el()),[key:"_onMouseenter",value:function(){this.show()},key:"_onMouseleave",value:function(){this.hide()}
8617 return t}])
8618 An.prototype.options ={name:"ControlTooltip"}  
8619
```

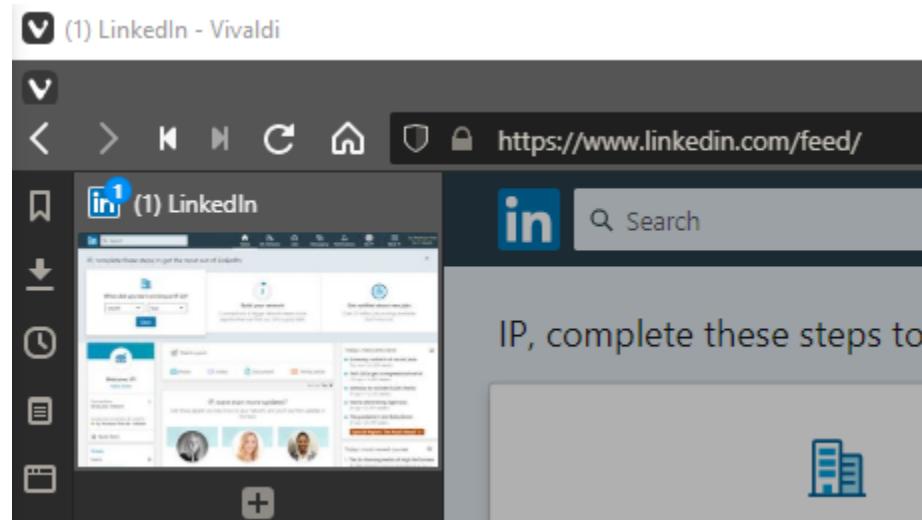
See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

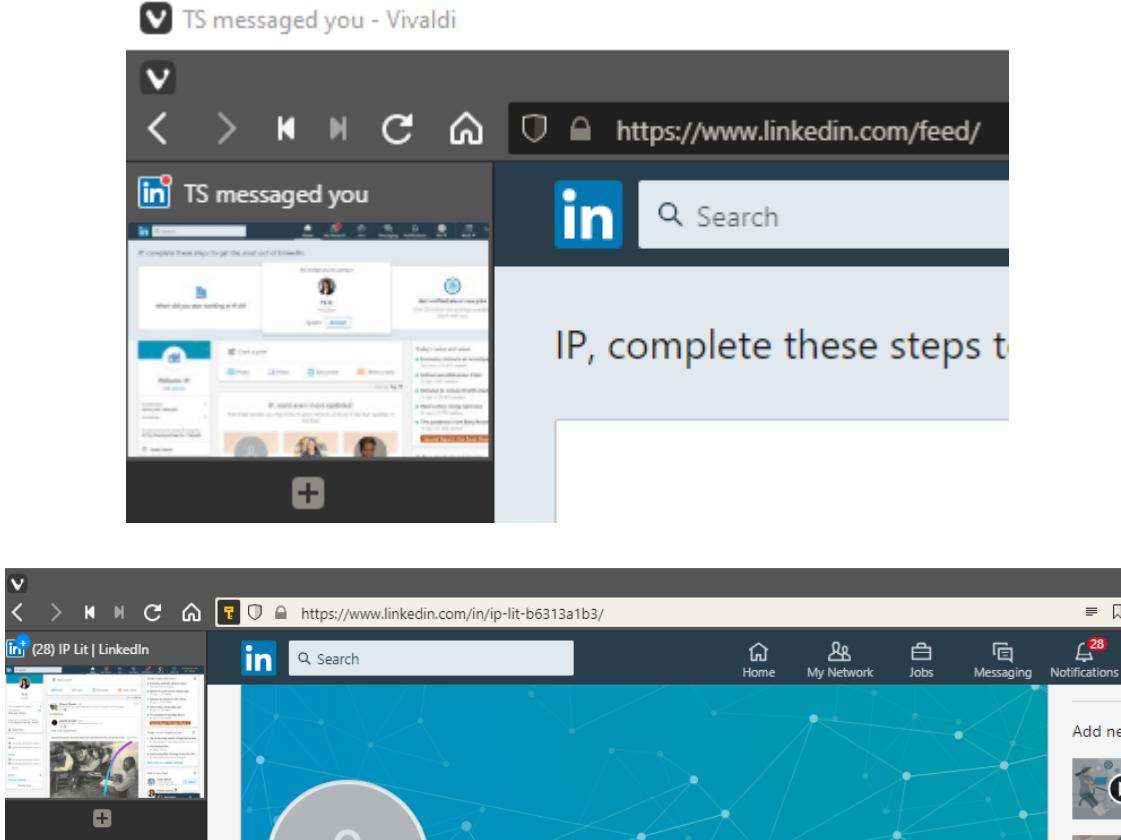
232. The LinkedIn Application provides an alternative title from the array to the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the user's browser with titles for each of the entries in the titlebar array, which is sent to the processor of the user's device for updating the HTML run and displayed by the user's browser:



```
23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]]else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=0
23404 this.pollTask("_loopTitles","","concat("document.title-poller","_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){re
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)})
23413 e.default=n}
23414 define("image-edit-base/components/bg-image-edit", ["exports", "image-edit-base/components/picture-cropper", "image-edit
```

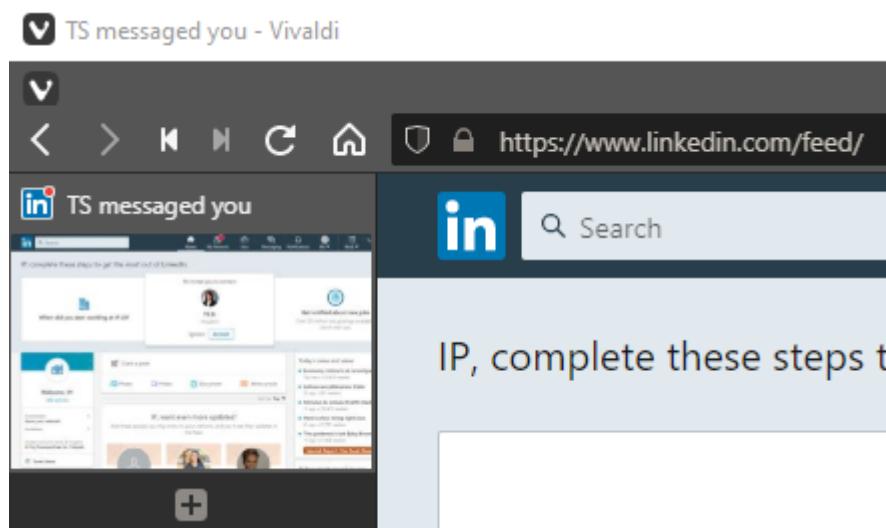
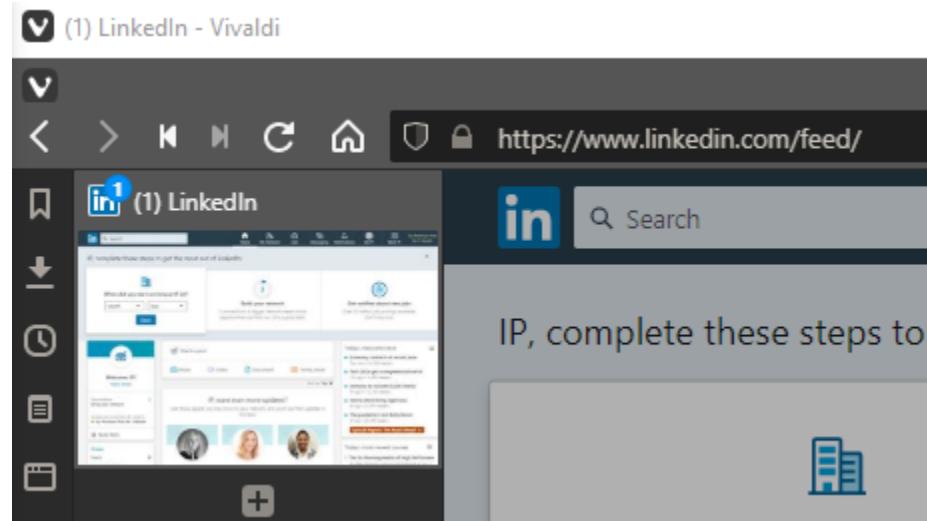
```
2112 void o==({n:e,t}||{t in e}) function f=typeof e.unknownProperty||(n.e.unknownProperty(t))
2113 if(a&&ee()){Array.isArray(n)||(!o.r.isEmberProxy)(n)&&Z(ie(n,"[ ])||(o,r.isProxy)(n)&&Z(ie(n,"content")))
2114 n=S(e,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i), a = {enumerable: true, configurable: true, get: f, set: f}
2116 ●if(null!=a)●return Ne(o,a,n) o = "LinkedIn", n = "(1) LinkedIn"
2117 if(r)throw new l.default("Property set failed: object in path '"+i.join(".")+" could not be found.")}(e,t,n,i) i
2118 var o,a=(o,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof o||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&e(t,e,t)e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(o.t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n. readOnly=!1
```





See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed/>

233. The LinkedIn Application uses the alternative title as a title in association with the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the user's browser with titles for each of the entries in the titlebar array, which is sent to the processor of the user's device for updating the HTML run and displayed by the user's browser:



```
23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)}  
23398 this._titles=[]  
23399 this._loopNumber=0  
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc  
23401 this._titles=[e,this._defaultDocumentTitle]else this._titles.unshift(e)  
23402 this._titles=this._titles.uniq()  
23403 if(!this._pollerEnabled){this._pollerEnabled=!0  
23404 this.pollTask("loopTitles","","concat("document-title-poller","").concat(Date.now()))},resetTitle:function(){if(t  
23405 this._loopNumber=0  
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle  
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r  
23408 this._titles[n-1]=this._defaultDocumentTitle  
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"  
23410 t.default&&a&&(document.title=a)  
23411 this._loopNumber++  
23412 this._pollerEnabled&&this.runTask(e,1500)})}  
23413 e.default=n)  
23414 define("image-edit-base/components/bc-img-edit","extends","image-edit-base/components/picture-converter","image-edit
```

```
2112 void e||(n=E(t))&&e[t] in e||function i=typeof e.unKnownProperty||!(n=e.unknownProperty(t))
2113 if(a&&ee){Z(e,e,t);(Array.isArray(n))&&(o.r.isEmberArray)(n))&&Z(i[e,n,""]);@,r.isProxy)(n)&&Z(i[e,n,"content"])
2114 n=S(e,n,r[i])return n}function Ne(e,t,n,i){if(e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i), a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)● return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'"' could not be found.'))(e,t,n,i)
2118 var o,a=(o,r.lookupDescriptor)(e,t),s=null==a:void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!=s&&q.has(s)[e[t]=n = " (1) LinkedIn"
2120 return n}if(void 0!=o[e[t]]||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&Le(e,t))else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){@,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))|this._volatile=1
2124 n._readOnly=i
```

```
2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"==i,a=0||"function"==i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0===(n=e[t])&&(o||t in e||"function"!=typeof e.unknownProperty()||(n=e.unknownProperty(t)))
2113 if(a&&ee(t)){(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(0,r.isProxy)(n)&&Z(ie(n,"content"))}return n}functi
2114 n=Se(n,r[i])&&return n}function Ee(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e =
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 0 if(null!=a)● return Ne(a,o,n)o = "(1) LinkedIn", n = "LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+' could not be found.')})(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.updateObject)(e,t),s=null==a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!=s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n}if(void 0!=-(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty(){e[t]=n o = "(1) LinkedIn"
2121 o!=null&&Pe(e,t))else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){0,t.inheritsLoose)(i,e)
2123 function i(t){var n=(e.call(this)||this).__volatile!=1
2124 n = n?0:1;n+=1
2125 }
```

```
1890 c=(function(){if(c){n  
1891 return!Ember.test;  
1892 u.call(e,c)  
1893 return m}  
1894 e.cancelPolls=  
1895 e.queuedPollTasks=  
1896 var a=new WeakMap  
1897 var o,r=0  
1898 var i=Object.create  
1899 e.queuedPollTasks=  
1900 function s(e,t){vi  
1901 if("number"==type(  
1902 else{var o=a.get(  
1903 n=t  
1904 void !==o&&o.del  
1905 define("ember-lifeline",  
1906 Object.defineProperty(  
1907 e,_setRegisteredTask=function(e){i=e  
1908 e.runTask=function([t]{var o=arguments.length>2&&void !==arguments[2]?arguments[2]:0  
1909 if(e.isDestroying) return a  
1910 var r=(o,n.default)(e,t,"runTask"),s=i(e),l=Ember.run.later((function(){s.delete(l)  
1911 r.call(e)},o)  
1912 s.add(l)  
1913 s.notifyAll()  
1914 })})});
```

```
2101 u||!e==Object.defineProperty(e,t,{configurable:!0,enumerable:c,writable:!0,value:l});eft!i}else{t=Object.defineProperty(e,t,r).o.isPrototypeOfMeta  
2102 t.lastRevision=(0,s.value(t.tag))  
2103 A.has(e)&&A.get(e).forEach((function(t){t.tag=t.lastRevision=(0,s.value(t.tag))}))}(e)  
2104 "function"==typeof e.didDefineProperty&e.didDefineProperty  
2105 function Ae(e){return"string"==typeof e&&e!=e.PROXY_CONTENT?e:  
2106 e.  
2107 0  
2108 function Se(e,t){var n,i=typeof e,o="object"==  
2109 if(Ae(t))return a?Pe(e,t):void 0  
2110 void 0===(n=e[t])&&(o||t in e)"function"!=typ  
2111 if(a&&e[i])Z(i,e[t]);Array.isArray(n)||(!0,r  
2112 n=S(e,n,r[i]))return n}function Ne(e,t,n,i){if(  
2113 var a=Pe(e,i) a = {enumerable: true, configur  
2114 if(null!=a) return Ne(a,o,n) o = "(1) LinkedIn  
2115 if(!r)throw new L.default("Property set failed:  
2116 var o={0,r.lookupDescriptor}(e,t),s=null;r  
2117 if(void 0!==s&&s.has(qs)){e[t]=n = n||(1) Link  
2118 2200 r||if(void 0===(o=e[t])||"object"!=typeof  
2119 o||n!=e[t])else e.setUnknownProperty(t,n  
2120 2201 o=n&e[e.t]||e.setUnknownProperty(t,n  
2121 2202 2203 2204 2205 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 2216 2217 2218 2219 2220 2221 2222 2223 2224 2225 2226 2227 2228 2229 2230 2231 2232 2233 2234 2235 2236 2237 2238 2239 2240 2241 2242 2243 2244 2245 2246 2247 2248 2249 2250 2251 2252 2253 2254 2255 2256 2257 2258 2259 2260 2261 2262 2263 2264 2265 2266 2267 2268 2269 2270 2271 2272 2273 2274 2275 2276 2277 2278 2279 2280 2281 2282 2283 2284 2285 2286 2287 2288 2289 2290 2291 2292 2293 2294 2295 2296 2297 2298 2299 2300 2301 2302 2303 2304 2305 2306 2307 2308 2309 2310 2311 2312 2313 2314 2315 2316 2317 2318 2319 2320 2321 2322 2323 2324 2325 2326 2327 2328 2329 2330 2331 2332 2333 2334 2335 2336 2337 2338 2339 2340 2341 2342 2343 2344 2345 2346 2347 2348 2349 2350 2351 2352 2353 2354 2355 2356 2357 2358 2359 2360 2361 2362 2363 2364 2365 2366 2367 2368 2369 2370 2371 2372 2373 2374 2375 2376 2377 2378 2379 2380 2381 2382 2383 2384 2385 2386 2387 2388 2389 2390 2391 2392 2393 2394 2395 2396 2397 2398 2399 2400 2401 2402 2403 2404 2405 2406 2407 2408 2409 2410 2411 2412 2413 2414 2415 2416 2417 2418 2419 2420 2421 2422 2423 2424 2425 2426 2427 2428 2429 2430 2431 2432 2433 2434 2435 2436 2437 2438 2439 2440 2441 2442 2443 2444 2445 2446 2447 2448 2449 2450 2451 2452 2453 2454 2455 2456 2457 2458 2459 2460 2461 2462 2463 2464 2465 2466 2467 2468 2469 2470 2471 2472 2473 2474 2475 2476 2477 2478 2479 2480 2481 2482 2483 2484 2485 2486 2487 2488 2489 2490 2491 2492 2493 2494 2495 2496 2497 2498 2499 2500 2501 2502 2503 2504 2505 2506 2507 2508 2509 2510 2511 2512 2513 2514 2515 2516 2517 2518 2519 2520 2521 2522 2523 2524 2525 2526 2527 2528 2529 2530 2531 2532 2533 2534 2535 2536 2537 2538 2539 2540 2541 2542 2543 2544 2545 2546 2547 2548 2549 2550 2551 2552 2553 2554 2555 2556 2557 2558 2559 2560 2561 2562 2563 2564 2565 2566 2567 2568 2569 2570 2571 2572 2573 2574 2575 2576 2577 2578 2579 2580 2581 2582 2583 2584 2585 2586 2587 2588 2589 2589 2590 2591 2592 2593 2594 2595 2596 2597 2598 2599 2599 2600 2601 2602 2603 2604 2605 2606 2607 2608 2609 2609 2610 2611 2612 2613 2614 2615 2616 2617 2618 2619 2619 2620 2621 2622 2623 2624 2625 2626 2627 2628 2629 2629 2630 2631 2632 2633 2634 2635 2636 2637 2638 2639 2639 2640 2641 2642 2643 2644 2645 2646 2647 2648 2649 2649 2650 2651 2652 2653 2654 2655 2656 2657 2658 2659 2659 2660 2661 2662 2663 2664 2665 2666 2667 2668 2669 2669 2670 2671 2672 2673 2674 2675 2676 2677 2678 2679 2679 2680 2681 2682 2683 2684 2685 2686 2687 2688 2689 2689 2690 2691 2692 2693 2694 2695 2696 2697 2698 2699 2699 2700 2701 2702 2703 2704 2705 2706 2707 2708 2709 2709 2710 2711 2712 2713 2714 2715 2716 2717 2718 2719 2719 2720 2721 2722 2723 2724 2725 2726 2727 2728 2729 2729 2730 2731 2732 2733 2734 2735 2736 2737 2738 2739 2739 2740 2741 2742 2743 2744 2745 2746 2747 2748 2749 2749 2750 2751 2752 2753 2754 2755 2756 2757 2758 2759 2759 2760 2761 2762 2763 2764 2765 2766 2767 2768 2769 2769 2770 2771 2772 2773 2774 2775 2776 2777 2778 2779 2779 2780 2781 2782 2783 2784 2785 2786 2787 2788 2789 2789 2790 2791 2792 2793 2794 2795 2796 2797 2798 2798 2799 2799 2800 2801 2802 2803 2804 2805 2806 2807 2808 2809 2809 2810 2811 2812 2813 2814 2815 2816 2817 2818 2819 2819 2820 2821 2822 2823 2824 2825 2826 2827 2828 2829 2829 2830 2831 2832 2833 2834 2835 2836 2837 2838 2839 2839 2840 2841 2842 2843 2844 2845 2846 2847 2848 2849 2849 2850 2851 2852 2853 2854 2855 2856 2857 2858 2859 2859 2860 2861 2862 2863 2864 2865 2866 2867 2868 2869 2869 2870 2871 2872 2873 2874 2875 2876 2877 2878 2879 2879 2880 2881 2882 2883 2884 2885 2886 2887 2888 2889 2889 2890 2891 2892 2893 2894 2895 2896 2897 2898 2898 2899 2899 2900 2901 2902 2903 2904 2905 2906 2907 2908 2909 2909 2910 2911 2912 2913 2914 2915 2916 2917 2918 2919 2919 2920 2921 2922 2923 2924 2925 2926 2927 2928 2929 2929 2930 2931 2932 2933 2934 2935 2936 2937 2938 2939 2939 2940 2941 2942 2943 2944 2945 2946 2947 2948 2949 2949 2950 2951 2952 2953 2954 2955 2956 2957 2958 2959 2959 2960 2961 2962 2963 2964 2965 2966 2967 2968 2969 2969 2970 2971 2972 2973 2974 2975 2976 2977 2978 2979 2979 2980 2981 2982 2983 2984 2985 2986 2987 2988 2989 2989 2989 2990 2991 2992 2993 2994 2995 2996 2997 2998 2998 2999 2999 3000 3001 3002 3003 3004 3005 3006 3007 3008 3009 3009 3010 3011 3012 3013 3014 3015 3016 3017 3018 3019 3019 3020 3021 3022 3023 3024 3025 3026 3027 3028 3029 3029 3030 3031 3032 3033 3034 3035 3036 3037 3038 3039 3039 3039 3040 3041 3042 3043 3044 3045 3046 3047 3048 3049 3049 3049 3050 3051 3052 3053 3054 3055 3056 3057 3058 3059 3059 3060 3061 3062 3063 3064 3065 3066 3067 3068 3069 3069 3069 3070 3071 3072 3073 3074 3075 3076 3077 3078 3079 3079 3080 3081 3082 3083 3084 3085 3086 3087 3088 3089 3089 3089 3090 3091 3092 3093 3094 3095 3096 3097 3098 3098 3098 3099 3099 3099 3100 3101 3102 3103 3104 3105 3106 3107 3108 3109 3109 3110 3111 3112 3113 3114 3115 3116 3117 3118 3119 3119 3120 3121 3122 3123 3124 3125 3126 3127 3128 3129 3129 3129 3130 3131 3132 3133 3134 3135 3136 3137 3138 3139 3139 3139 3140 3141 3142 3143 3144 3145 3146 3147 3148 3149 3149 3149 3150 3151 3152 3153 3154 3155 3156 3157 3158 3159 3159 3160 3161 3162 3163 3164 3165 3166 3167 3168 3169 3169 3169 3170 3171 3172 3173 3174 3175 3176 3177 3178 3179 3179 3179 3180 3181 3182 3183 3184 3185 3186 3187 3188 3189 3189 3189 3190 3191 3192 3193 3194 3195 3196 3197 3198 3198 3198 3199 3199 3199 3200 3201 3202 3203 3204 3205 3206 3207 3208 3209 3209 3210 3211 3212 3213 3214 3215 3216 3217 3218 3219 3219 3220 3221 3222 3223 3224 3225 3226 3227 3228 3229 3229 3229 3230 3231 3232 3233 3234 3235 3236 3237 3238 3239 3239 3239 3240 3241 3242 3243 3244 3245 3246 3247 3248 3249 3249 3249 3250 3251 3252 3253 3254 3255 3256 3257 3258 3259 3259 3259 3260 3261 3262 3263 3264 3265 3266 3267 3268 3269 3269 3269 3270 3271 3272 3273 3274 3275 3276 3277 3278 3279 3279 3279 3280 3281 3282 3283 3284 3285 3286 3287 3288 3289 3289 3289 3290 3291 3292 3293 3294 3295 3296 3297 3298 3298 3298 3299 3299 3299 3300 3301 3302 3303 3304 3305 3306 3307 3308 3309 3309 3310 3311 3312 3313 3314 3315 3316 3317 3318 3319 3319 3320 3321 3322 3323 3324 3325 3326 3327 3328 3329 3329 3329 3330 3331 3332 3333 3334 3335 3336 3337 3338 3339 3339 3339 3340 3341 3342 3343 3344 3345 3346 3347 3348 3349 3349 3349 3350 3351 3352 3353 3354 3355 3356 3357 3358 3359 3359 3359 3360 3361 3362 3363 3364 3365 3366 3367 3368 3369 3369 3369 3370 3371 3372 3373 3374 3375 3376 3377 3378 3379 3379 3379 3380 3381 3382 3383 3384 3385 3386 3387 3388 3389 3389 3389 3390 3391 3392 3393 3394 3395 3396 3397 3398 3398 3398 3399 3399 3399 3400 3401 3402 3403 3404 3405 3406 3407 3408 3409 3409 3410 3411 3412 3413 3414 3415 3416 3417 3418 3419 3419 3420 3421 3422 3423 3424 3425 3426 3427 3428 3429 3429 3429 3430 3431 3432 3433 3434 3435 3436 3437 3438 3439 3439 3439 3440 3441 3442 3443 3444 3445 3446 3447 3448 3449 3449 3449 3450 3451 3452 3453 3454 3455 3456 3457 3458 3459 3459 3459 3460 3461 3462 3463 3464 3465 3466 3467 3468 3469 3469 3469 3470 3471 3472 3473 3474 3475 3476 3477 3478 3479 3479 3479 3480 3481 3482 3483 3484 3485 3486 3487 3488 3489 3489 3489 3490 3491 3492 3493 3494 3495 3496 3497 3498 3498 3498 3499 3499 3499 3500 3501 3502 3503 3504 3505 3506 3507 3508 3509 3509 3510 3511 3512 3513 3514 3515 3516 3517 3518 3519 3519 3520 3521 3522 3523 3524 3525 3526 3527 3528 3529 3529 3529 3530 3531 3532 3533 3534 3535 3536 3537 3538 3539 3539 3539 3540 3541 3542 3543 3544 3545 3546 3547 3548 3549 3549 3549 3550 3551 3552 3553 3554 3555 3556 3557 3558 3559 3559 3559 3560 3561 3562 3563 3564 3565 3566 3567 3568 3569 3569 3569 3570 3571 3572 3573 3574 3575 3576 3577 3578 3579 3579 3579 3580 3581 3582 3583 3584 3585 3586 3587 3588 3589 3589 3589 3590 3591 3592 3593 3594 3595 3596 3597 3598 3598 3598 3599 3599 3599 3600 3601 3602 3603 3604 3605 3606 3607 3608 3609 3609 3610 3611 3612 3613 3614 3615 3616 3617 3618 3619 3619 3619 3620 3621 3622 3623 3624 3625 3626 3627 3628 3629 3629 3629 3630 3631 3632 3633 3634 3635 3636 3637 3638 3639 3639 3639 3640 3641 3642 3643 3644 3645 3646 3647 3648 3649 3649 3649 3650 3651 3652 3653 3654 3655 3656 3657 3658 3659 3659 3659 3660 3661 3662 3663 3664 3665 3666 3667 3668 3669 3669 3669 3670 3671 3672 3673 3674 3675 3676 3677 3678 3679 3679 3679 3680 3681 3682 3683 3684 3685 3686 3687 3688 3689 3689 3689 3690 3691 3692 3693 3694 3695 3696 3697 3698 3698 3698 3699 3699 3699 3700 3701 3702 3703 3704 3705 3706 3707 3708 3709 3709 3710 3711 3712 3713 3714 3715 3716 3717 3718 3719 3719 3719 3720 3721 3722 3723 3724 3725 3726 3727 3728 3729 3729 3729 3730 3731 3732 3733 3734 3735 3736 3737 3738 3739 3739 3739 3740 3741 3742 3743 3744 3745 3746 3747 3748 3749 3749 3749 3750 3751 3752 3753 3754 3755 3756 3757 3758 3759 3759 3759 3760 3761 3762 3763 3764 3765 3766 3767 3768 3769 3769 3769 3770 3771 3772 3773 3774 3775 3776 3777 3778 3779 3779 3779 3780 3781 3782 3783 3784 3785 3786 3787 3788 3789 3789 3789 3790 3791 3792 3793 3794 3795 3796 3797 3798 3798 3798 3799 3799 3799 3800 3801 3802 3803 3804 3805 3806 3807 3808 3809 3809 3810 3811 3812 3813 3814 3815 3816 3817 3818 3819 3819 3819 3820 3821 3822 3823 3824 3825 3826 3827 3828 3829 3829 3829 3830 3831 3832 3833 3834 3835 3836 3837 3838 3839 3839 3839 3840 3841 3842 3843 3844 3845 3846 3847 3848 3849 3849 3849 3850 3851 3852 3853 3854 3855 3856 3857 3858 3859 3859 3859 3860 3861 3862 3863 3864 3865 3866 3867 3868 3869 3869 3869 3870 3871 3872 3873 3874 3875 3876 3877 3878 3879 3879 3879 3880 3881 3882 3883 3884 3885 3886 3887 3888 3889 3889 3889 3890 3891 3892 3893 3894 3895 3896 3897 3898 3898 3898 3899 3899 3899 3900 3901 3902 3903 3904 3905 3906 3907 3908 3909 3909 3910 3911 3912 3913 3914 3915 3916 3917 3918 3919 3919 3919 3920 3921 3922 3923 3924 3925 3926 3927 3928 3929 3929 3929 3930 3931 3932 3933 3934 3935 3936 3937 3938 3939 3939 3939 3940 3941 3942 3943 3944 3945 3946 3947 3948 3949 3949 3949 3950 3951 3952 3953 3954 3955 3956 3957 3958 3959 3959 3959 3960 3961 3962 3963 3964 3965 3966 3967 3968 3969 3969 3969 3970 3971 3972 3973 3974 3975 3976 3977 3978 3979 3979 3979 3980 3981 3982 3983 3984 3985 3986 3987 3988 3989 3989 3989 3990 3991 3992 3993 3994 3995 3996 3997 3998 3998 3998 3999 3999 3999 4000 4001 4002 4003 4004 4005 4006 4007 4008 4009 4009 4010 4011 4012 4013 4014 4015 4016 4017 4018 4019 4019 4019 4020 4021 4022 4023 4024 4025 4026 4027 4028 4029 4029 4029 4030 4031 4032 4033 4034 4035 4036 4037 4038 4039 4039 4039 4040 4041 4042 4043 4044 4045 4046 4047 4048 4049 4049 4049 4050 4051 4052 4053 4054 4055 4056 4057 4058 4059 4059 4059 4060 4061 4062 4063 4064 4065 4066 4067 4068 4069 4069 4069 4070 4071 4072 4073 4074 4075 4076 4077 4078 4079 4079 4079 4080 4081 4082 4083 4084 4085 4086 4087 4088 4089 4089 4089 4090 4091 4092 4093 4094 4095 4096 4097 4098 4098 4098 4099 4099 4099 4100 4101 4102 4103 4104 4105 4106 4107 4108 4109 4109 4110 4111 4112 4113 4114 4115 4116 4117 4118 4119 4119 4119 4120 4121 4122 4123 4124 4125 4126 4127 4128 4129 4129 4129 4130 4131 4132 4133 4134 4135 4136 4137 4138 4139 4139 4139 4140 4141 4142 4143 4144 4145 4146 4147 4148 4149 4149 4149 4150 4151 4152 4153 4154 4155 4156 4157 4158 4159 4159 4159 4160 4161 4162 4163 4164 4165 4166 4167 4168 4169 4169 4169 4170 4171 4172 4173 4174 4175 4176 4177 4178 4179 4179 4179 4180 4181 4182 4183 4184 4185 4186 4187 4188 4189 4189 4189 4190 4191 4192 4193 4194 4195 4196 4196 4196 4197 4197 4197 4198 4198 4198 4199 4199 4199 4200 4201 4202 4203 4204 4205 4206 4207 4208 4209 4209 4210 4211 4212 4213 4214 4215 4216 4217 4218 4219 4219 4219 4220 4221 4222 4223 4224 4225 4226 4227 4228 4229 4229 4229 4230 4231 4232 4233 4234 4235 4236 4237 4238 4239 4239 4239 4240 4241 4242 4243 4244 4245 4246 4247 4248 4249 4249 4249 4250 4251 4252 4253 4254 4255 4256 4257 4258 4259 4259 4259 4260 4261 4262 4263 4264 4265 4266 4267 4268 4269 4269 4269 
```

```

2101 u||!1==c?Object.defineProperty(e,t,{configurable:!0,enumerable:c,writable:!0,value:l}):e[t]=i}else{l=r
2102 Object.defineProperty(e,t,r)o.isPrototypeOf(e)||((function(e){C.has(e)&&C.get(e).forEach((function(t){t.tag=(0,s.combine)(we(e,t,p
2103 t.lastRevision=(0,s.value)(t.tag)))}
2104 A.has(e)&&A.get(e).forEach((function(t){t.tag=(0,s.combine)(we(e,t.path))
2105 t.lastRevision=(0,s.value)(t.tag))))}(e)
2106 "function"==typeof e.didDefineProperty&&e.didDefineProperty(e,t,l){var ke=new r.Cache(1e3,function(e){return e.indexOf(".")})
2107 function Ae(e){return"string"==typeof e&&1==ke.get(e)}var Ce=(0,r.symbol)("PROXY_CONTENT")
2108 e.PROXY_CONTENT=ce
2109 }
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a=o|"function"==i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(o|t in e||"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&ee()||(Z(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"["]));(!0,r.isProxy)(n)&&Z(ie(n,"content"))))return n}functi
2114 n=Se(n,r[i]);return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,i){var i=t.split("."),o=i.pop() e=i
2115 var a=Pe(e,i) a={enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)return Ne(a,o,n) o = "(1) LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default("Property set failed: object in path "+i.join(".")+" could not be found.")(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?vo["title"] : o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n m = "(1) LinkedIn"
2120 return n}if(void 0==(o=e[t])||"object"!=typeof e[t]||in e["function"!=typeof e.setUnknownProperty]{e[t]=n
2121 o=n&&e[e.t])else e.setUnknownProperty(t,n)

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed/>.

234. LinkedIn has directly infringed, and continues to directly infringe, the claims of the ‘179 Patent, including at least those noted above, including by at least making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a).

235. LinkedIn has had at least constructive notice of the ‘179 Patent since at least its issuance. LinkedIn will have been on actual notice of the ‘179 Patent since, at the latest, the service of this complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the ‘179 Patent.

236. EBT believes and contends that, at minimum, LinkedIn’s knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the ‘179 Patent since receiving notice of its infringement of the ‘179 Patent, is necessarily willful, wanton, malicious, in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a finding of willful infringement. Accordingly, since at least receiving notice of this suit, LinkedIn has willfully infringed the ‘179 Patent.

## DAMAGES

237. By way of its infringing activities, LinkedIn has caused, and continues to cause, Plaintiff to suffer damages, and Plaintiff is entitled to recover from LinkedIn the damages sustained by Plaintiff as a result of LinkedIn’s wrongful acts in an amount subject to proof at

trial, which, by law, cannot be less than a reasonable royalty, together with interest and costs as fixed by this Court under 35 U.S.C. § 284.

238. LinkedIn's infringement of Plaintiff's rights under the Patents-in-Suit will continue to damage Plaintiff, causing irreparable harm for which there is no adequate remedy at law, unless enjoined by this Court.

239. Plaintiff also requests that the Court make a finding that this is an exceptional case entitling Plaintiff to recover its attorneys' fees and costs pursuant to 35 U.S.C. § 285.

**DEMAND FOR JURY TRIAL**

240. Pursuant to Rule 38 of the Federal Rules of Civil Procedure, Plaintiff hereby respectfully requests a trial by jury of any issues so triable by right.

**PRAYER FOR RELIEF**

WHEREFORE, EBT hereby respectfully requests that this Court enter judgment in favor of EBT and against Defendant, and that the Court grant EBT the following relief:

- A. Judgment that Defendant has infringed and is infringing the Patents-in-Suit;
- B. Judgment that LinkedIn's post-notice infringement has been, and continues to be, willful, including that LinkedIn acted to infringe the Patents-in-Suit despite an objectively high likelihood that its actions constituted infringement of a valid patent and, accordingly, award enhanced damages, including treble damages, pursuant to 35 U.S.C. § 284;
- C. An award to Plaintiff of damages adequate to compensate Plaintiff for LinkedIn's infringement, together with pre-judgment and post-judgment interest, and any continuing or future infringement through the date such judgment is entered, including interest, costs, expenses, and an accounting of all infringing acts including, but not limited to, those acts not presented at trial;
- D. A grant of a permanent injunction pursuant to 35 U.S.C. § 283, enjoining LinkedIn and

all persons, including its officers, directors, agents, servants, affiliates, employees, divisions, branches, subsidiaries, parents, and all others acting in active concert or participation therewith, from making, using, offering to sell, or selling in the United States or importing into the United States any methods, systems, or computer readable media that directly or indirectly infringe any claim of the Patents-in-Suit, or any methods, systems, or computer readable media that are colorably different;

- E. That this Court declare this to be an exceptional case and award Plaintiff reasonable attorneys' fees and costs in accordance with 35 U.S.C. § 285; and
- F. Any and all further relief for which Plaintiff may show itself justly entitled that this Court deems just and proper.

Dated: February 25, 2021

Respectfully submitted,

Of Counsel:

Steve Schlather  
EDMONDS & SCHLATHER PLLC  
2501 Saltus St.  
Houston, Texas 77003  
Phone: (713) 364-2371  
Fax: (713)224-6651  
sschlather@ip-lit.com

FARNAN LLP

/s/ Brian E. Farnan  
Brian E. Farnan (Bar No. 4089)  
Michael J. Farnan (Bar No. 5165)  
919 Market St., 12<sup>th</sup> Floor  
Wilmington, DE 19801  
Telephone: (302) 777-0300  
Fax: (302) 777-0301  
bfarnan@farnanlaw.com  
mfarnan@farnanlaw.com

*Attorneys for Plaintiff*